Magyar számítástechnikai folyóirat CD-melléklettel Ára 699 Ft

XVIII. évfolyam 4. szám 2000. április

ÚJ ALAPLAP

A hónap témája:

HUNOSÍTÁS



Netszabadság

Nyílt tér

Videómánia

Közkincs

A partíciók varázslója

Fogódzó

Fordulat a vírusfronton I.

Vírusőrjárat

Számítógéppel emberi nyelven

Könyvespolc

00004

771217 759005



Ha én ezt a Klubban egyszer elmesélem...
Napi 8 óra internet ingyen!*
Elender Klub. Kultúra, szórakozás és információcsere a legjobb társaságban.
Engedje meg, hogy bemutassuk Önnek az internet legjavát! Előfizetőink most tagdíjfizetés nélkül látogathatják az Elender Klubot, ahol mindenki és minden ott van, aki és ami számít. Belépés csak fesztelenül!
Nézzen körül a képtárban, ismerkedjen meg a legfrissebb irodalmi alkotásokkal, böngéssze át internetes ajánlatainkat, mazsolázgasson az érdekes események között, faggassa ki a számítástechnikai szakértőket, találjon új barátokra vagy társalogjon ismeretlenekkel. Akár WAP-on is.
Az Elender Klubban egyetlen szabály van: érezze jól magát. Úgy, ahogy Önnek tetszik.
* Hétköznaponként 4 és 8 óra, valamint 16 és 20 óra között, hétvégenként pedig 8 és 16 óra között az Elender nem számlázza az interneten töltött időt.
EURO RSCG NEW EUROPE
INFORMÁCIÓKÉRÉS: 10 ▲
elender internet
PSINet™
AZ ELENDER A PSINET CSOPORT TAGJA
Elender Informatikai Rt. Tel.: 465-7800; Fax: 465-7899; 1134 Budapest, Váci út 37. www.eol.hu

ÚJ ALAPLAP

A Mikroszámítógép Magazin és az Alaplap hagyományait folytató magyar számítástechnikai folyóirat

Megjelenik havonta, CD-melléklettel

Főszerkesztő:
Faklen Pál

Szerkesztő:
Jakab Ágnes

A szerkesztőbizottság tagjai:
Aszalós László, Bánó György, Feleki Zoltán, Galántai Zoltán, Herczeg József, Kádár Zsolt, Kovács Attila, Mákos András, Nagy Tamás, Pogány Csaba, Sándor Gábor, Simay Endre István, Szappanos Gábor, Szondi Egon János, Vargha Dénes, Vékony Tamás

Szerkesztőség és kiadó:
1539 Budapest, Pf. 571
VI., Dózsa György út 84/b
Telefon: 322-4417, 322-5238
Fax: 351-8015
E-mail: alaplap@mail.datanet.hu
Weblap: http://www.alaplap.hu

Felelős kiadó:
Faklen Pál

Terjesztés:
Megyes Zsuzsanna

Hirdetésszervezés:
Árvai Katalin,
Galyasi Hedvig,
Pap Katalin

Külföldi hirdetések:
PubliCity
Reklám- és Médiaügynökség
1537 Budapest I., Márvány u. 17.
Telefon: 356-1182 Fax: 375-3539

A kiadó a hirdetések tartalmáért és a nyomdakészen kapott hirdetések formájáért (és helyesírásáért) nem vállal felelősséget

Példányszámadatok hitelesítése:
Magyar Terjesztésellenőrző Szövetség
MATESZ

Ez a szám
9000 példányban jelent meg

Nyomtatás:
Zalai Nyomda Rt, Zalaegerszeg
Felelős vezető:
Czirkl György vezérigazgató

Terjeszti:
a Lapker Rt, a Hírker Rt, az NH Rt, az MP Rt LHI és számos számítástechnikai szaküzlet

Előfizethető a kiadónál:
Új Alaplap Kiadói Kft,
1539 Budapest, Pf. 571
Bankszámlaszám:
OTP 11706016-20788599

A lap példányonkénti ára: 699 Ft
Évi előfizetési díj: 6996 Ft

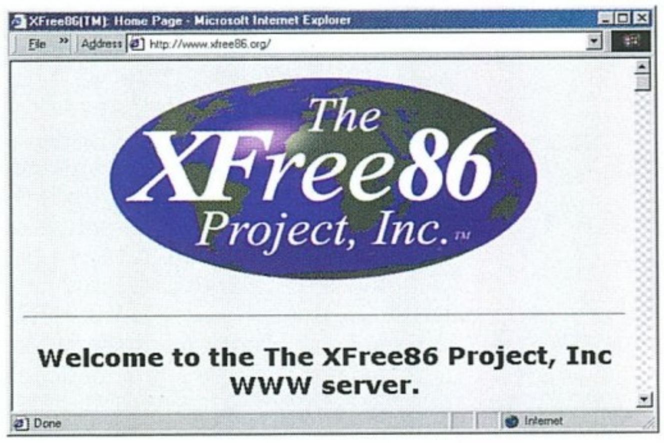
Külföldi előfizetés díja:
6996 Ft + postázási költség

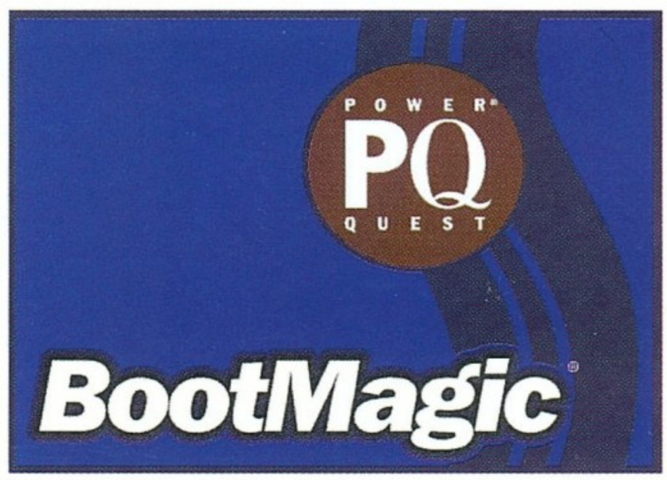
HU ISSN 1217-7598

TARTALOM

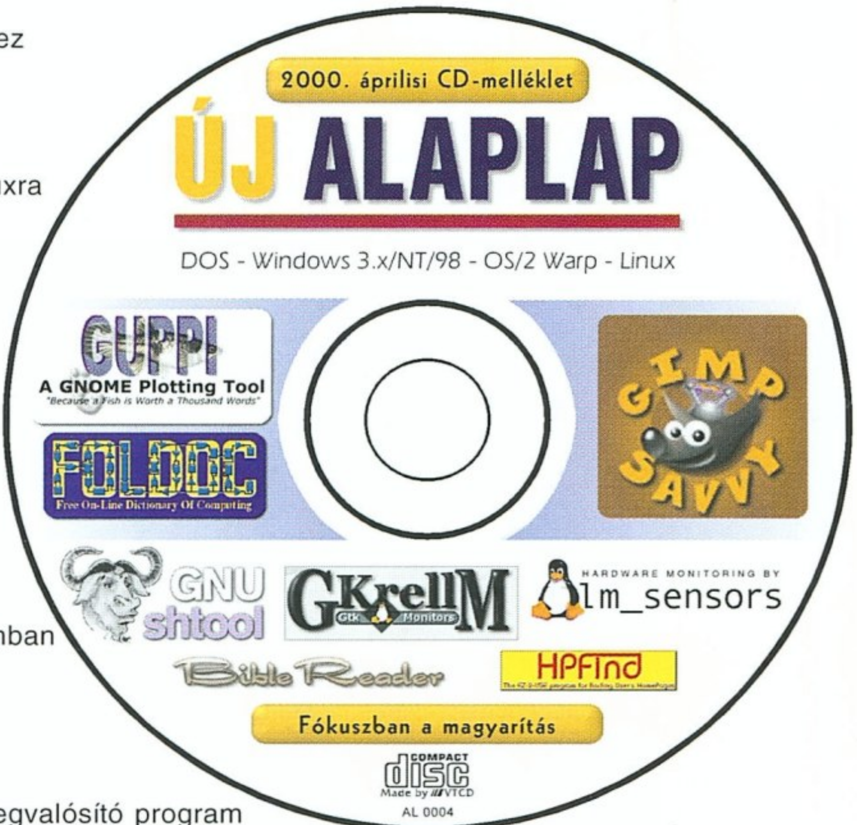
XVIII. ÉVFOLYAM 4. SZÁM, 2000. ÁPRILIS











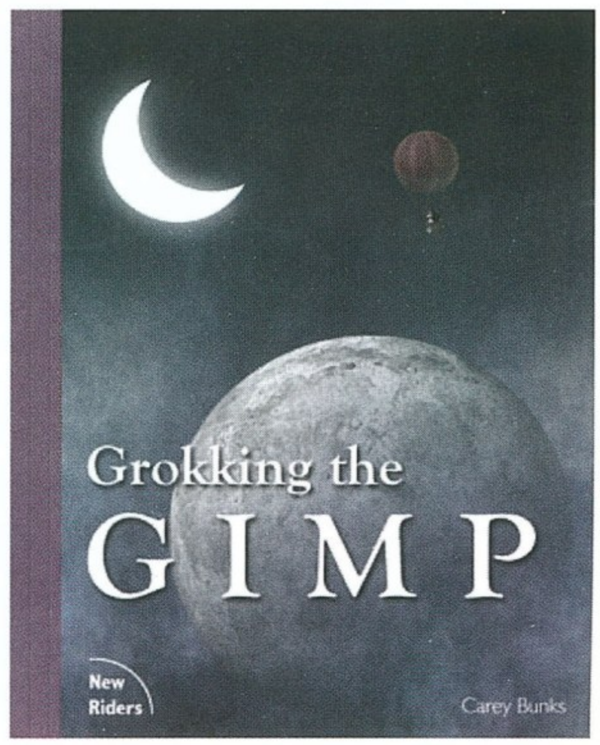
Fókuszban a magyarítás

FOKUSZ	
DODI	Egy honlap a programok magyarításáról, freeware/shareware magyarul
FONTAPPS	Betűkészleteket kezelő szofverek (nézegető, szerkesztő programok stb.)
	Font Creator v1.2.3 és v2.2, segédprogram TrueType fontok szerkesztéséhez
	Font Creator Tutorial, kézikönyv a Font Creator programhoz (HTML)
	FontLister 2.0 és 3.4.9, segédprogram betűkészletek telepítéséhez
	Font Impressions 2.0, segédprogram betűkészletek kezeléséhez
	FontView V7.0, segédprogram a telepített betűkészletek megtekintéséhez
	FontSelector 1.3, segédprogram a telepített betűkészletek megtekintéséhez
	Printers Apprentice 7.11, segédprogram betűkkészletek kezeléséhez
	Softy 107b, segédprogram betűk (fontok) készítéséhez
	TT Converter 1.5, PC-s betűkészleteket Mac-re konvertáló segédprogram
FREETYPE	FreeType library és fontszerver
FT1SRC	FreeType/1 library forráskód, betűkészletek kezeléséhez
FT2SRC	FreeType/2 fontszerver forráskód, betűkészletek kezeléséhez
FT2BIN	FreeType/2 fontszerver, bináris állományok OS/2-re
TTFCONV	TrueType fontkonverterek
GOMBKOD	Forráskód a 23. oldalon lévő cikkhez
JARGON	A Jargon File, Eric S. Raymondtól, HTML formátumban
KVOCTRAIN	Kvoctrain 0.6, idegen szavak tanulását segítő program Linuxra
SOKNYELV	Kétnyelvű szótárak (angol-magyar, magyar-angol stb.)
TECHSZOT	Számítástechnikai értelmező szótár TXT formátumban
TELESZO	A Teleszó hangfelismerő rendszer honlapja
UNI	Unicode 3.0
CHARTS	Unicode karaktertáblázatok (HTML és PDF)
FAQ	Unicode-dal kapcsolatos kérdések
GLOSSARY	Glossary, kifejezések magyarázata angolul (HTML)
UPDATE30	Unicode 3.0 karakter-adatbázis, Unihan adatbázis
YUDIT	Unicode-ot támogató szövegszerkesztő X Window-ra

Lapraforgó

LAPFORG	
JAVA	Illusztrációk a Java tanfolyamhoz (63. oldal)
JAVATANF	Példaprogramok a Java tanfolyam 4. részéhez
JAVATUT	The Java Tutorial, Java tankönyv a Sun-tól HTML formátumban
KOZKINCS	Videólejátszók különböző platformokra (40. oldal)
OS2	Nemcsak OS/2 ... (32. oldal)
DBASEINF	dBase Info 0.95, grafikus program DBF fájlok kezeléséhez
JFS	A JFS (Journaling File System) fájlrendszer forráskódja
LINKS	Links 0.82, karakteres böngészőprogram
LPTOOL	LPTool 2.11, párhuzamos portokon keresztüli adatcserét megvalósító program
NICE	Nice OS/2 Enhancer 3.1, segédprogram a munkaasztal csinosításához
RPM	RPM 3.0.3, Red Hat Package Manager OS/2-re
VMWARE	VMWare 2.0 béta, PC-emulátor különböző platformokra
URL	A lapban hivatkozott URL címek listája
WINLIN	Windowstól a szabad forráskódig (23. oldal)
	Javítócsomagok (patch) Windows 95-höz és 98-hoz





Szerszámosláda



SZERSZAM	
DOS	DOS-os alkalmazások
WIN3X	16 bites windowsos alkalmazások
WIN9X	32 bites windowsos alkalmazások
LINUX	Linuxos alkalmazások
OS2	OS/2-es alkalmazások
FESZER	Gyakran szükséges programok
VIRUS	Vírusirtók

Vendégoldal



VENDEG	
ARCH	AutoArchiver az InfoCentrumtól
BORLAND	Programok az Inprise-tól
CPP55FRE	Borland C++ Compiler 5.5 with Command Line Tools Version 5.5
VBJ40	Visibroker for Java 4.0
INTERBAS	InterBase 6.0 béta változat
DELCOMP	Delphi komponensek
GIMPBOOK	Grokking the Gimp, tankönyv a Gimpről
GULYAS	File Wizard (Gulyás Antal Dénes)
KKERESO	Kincskereső az Elendertől
MICROSFT	Programok, javítócsomagok a Microsofttól
ADSI25	Active Directory Service Interfaces v2.5
OFF97SR2	Office 97 Service Pack 2, magyar és angol változat
NETREXX	Kádár Zsolt NetRexx tanfolyama (1–5. rész)
NOVELL	NetWare kliens v4.7 Windows NT-re és Windows 2000-re
PAL	A galvánelem működését bemutató program (Pál László)
SAMBA	Samba 2.06 program és a Using Samba
STARLOGO	StarLogo 1.0, a Logo kibővített változata
SUN	Java SDK 1.2.2 Linuxra
XFREE86	XFree86 4.0, bináris és forráskód fájlok



Játékvár

JATEK	
CPLUS	CrosswordPlus 2.0, keresztrejtvénykészítő
PWORM	PizzaWorm 2.1, ügyességi játék
TABLA	Táblajátékosok fóruma



„Hunok” vagyunk

Amikor lapunkat, a Mikroszámítógép Magazint 10 évvel ezelőtt átalakítottuk Alaplappá, a hónap témája összeállításban elsőként a számítástechnika és a magyar nyelv kapcsolatát vettük nagyító alá (Magzarul besyelunk, Alaplap, 1990. június), és többször visszatértünk annak azóta is aktuális — vagy időközben azzá vált — kérdéseihez.

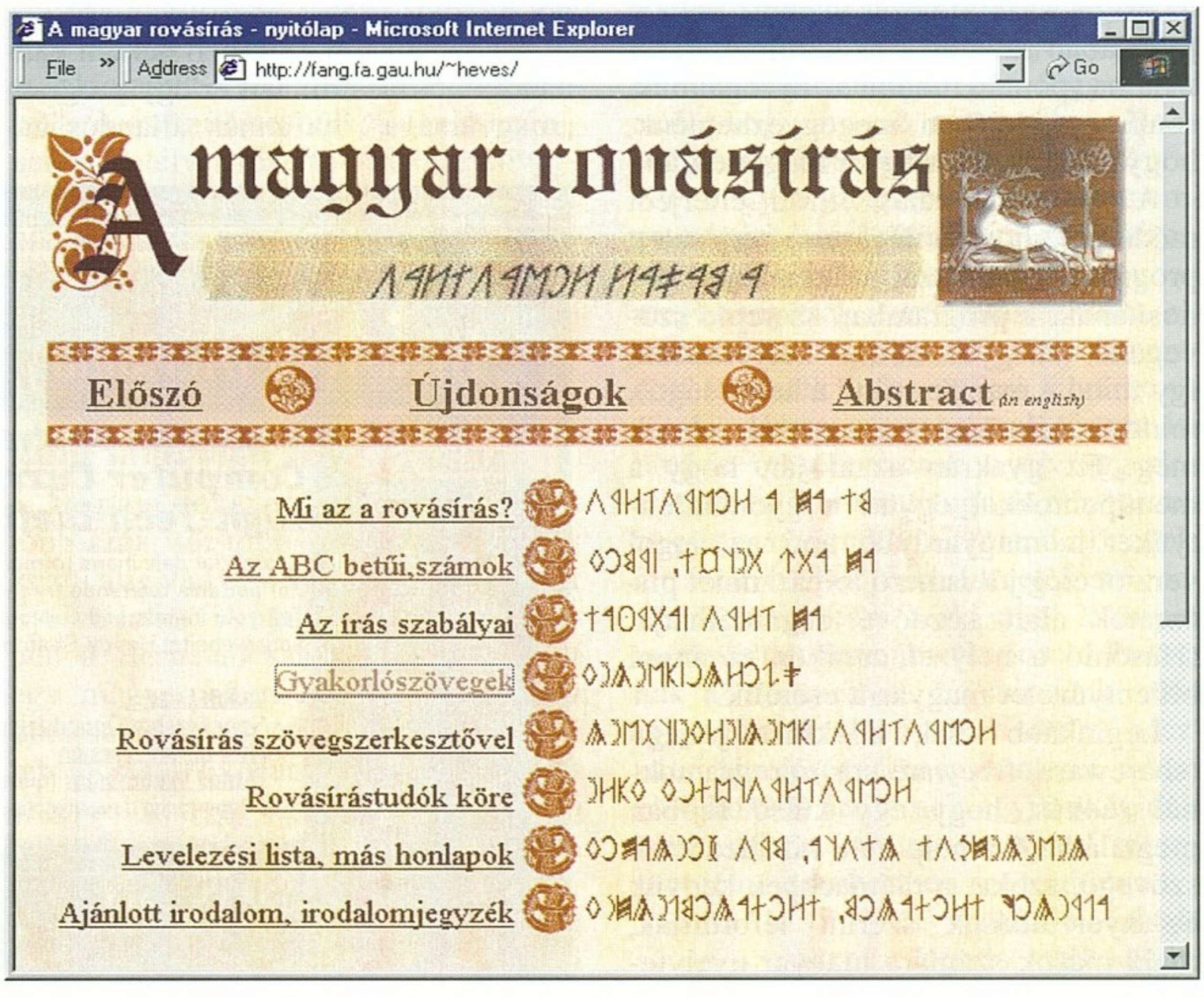
Jó lett volna, ha ez alatt a 10 év alatt az informatikai lap- és könyvkiadók, a hazai szoftverfejlesztő műhelyek, a nyelvművelő intézmények és más érdekeltek szorosabban együttműködtek volna e szaknyelv egységessé válása, a terminológia egyeztetése céljából, a (piaci) körülmények azonban nem kedveztek egy ilyen összefogásnak. A szaknyelv aktív formálói többnyire elszigetelten, saját eszközeikre hagyatkozva alakították ki saját megoldásaikat. Hosszabb távon persze figyelembe tudták venni (igenlően vagy tagadva) a másutt követett gyakorlatot, de ennyire dinamikusan fejlődő szakterületen az új fogalmak kezelése és egyértelmű jelölése nagyobb előzetes együttműködést is megkívánna. (Még nem késő!)

Naponta bukkannak fel a külföldi szakmai hírek között előzőleg soha nem hallott elnevezések. Valamilyen nyelvi lelemény mindenképpen szükséges magyar nyelvű ismertetésükhöz, mert amit meg kell magyarázni, arra az eredeti idézése mellett célszerű magyar elnevezést is találni, legyen az tükörfordítás, magyar analógia, körülírás, vagy ilyenek kombinációja. És ilyesmiben néha órák és napok alatt kell dönteni, mert nyakunkon a lapzárta... A szakújságíró kitalál valamit, de más lapokban ugyanazt másként magyarítják, és elindul egy hosszú folyamat.

10–15 évvel ezelőtt a külföldről importált új szakmai kifejezések magyar nyelvi környezetbe való átültetésének tűzvonalában még a Mikroszámítógép Magazin (Alaplap) és a Computerworld-Számítástechnika állt, később sorakozott fel mellénk a többi számítástechnikai folyóirat. Az utóbbi néhány évben a helyzet azonban alaposan megváltozott, mert az első megfogalmazás kényszere áttevődött az informatikai újdonságokat a napilapoknál is fürgébben publikáló webes médiákra (Internetto, Index, Prím Online, Terminal stb.). A többiek már a weben („első felindulásban”) kitalált szóhasználatra kénytelenek támaszkodni.

Az első névadás befolyásoló ereje kétségtelenül nagy, sok ideiglenes név a megszokás folytán véglegesen ráragad a fogalmakra, de azért a vitás esetekben sokkal inkább a nyomtatott szaklapoknál kialakuló gyakorlat szabja meg a magyarítás irányát. Ezzel a témával egyébként is nehéz úgy foglalkozni, hogy ne ütközzünk lépten-nyomon teljesen más felfogásba, véleménybe. (Bár ez nem olyan nagy baj, mint sokan gondolják.)

Végül, amit a honosítás szó a mi szakterületünkön általában kifejez, és amivel legtöbb cikkünk is foglalkozik, az a szoftverek átdolgozása magyar nyelvi környezetben való alkalmazásukhoz. Ezt a tevékenységet magyarításnak, fordításnak, lokalizációnak is nevezik. A hónap témája összefoglaló címéül nem azért választottuk a „hunosítást”, hogy a lehetséges elnevezések számát tovább szaporítsuk, és nem is azért, hogy a hun rokonság teóriájával kacérkodjunk. Mindössze arról van szó, hogy a külvilágban bennünket H, HU, HUN rövidítésekkel és azok teljes alakjával (Hungary, Hungaria stb.) jelölnek, kezdve az internetes címektől a sportversenyek eredményjelző tábláin át a hivatalos dokumentumokig. Ilyen játékos értelemben tehát „hunok” vagyunk. És akkor már az archaizálás kedvéért a magyar rovásírással foglalkozó honlap képét tettük ide.



Más dolog írni, más olvasni

A programoknak lehetne eszperantójuk

Joggal várható el, hogy a magyaroknak készített termékek „tudjanak magyarul". Lassanként már a háztartási gépek beállítása (installálása) sem egyszerű feladat, magyar nyelvű használati utasítás nélkül pedig nem is megy mindenkinek. Mióta a számítógép használata széles körben elterjedt, vele szemben is hasonló igényeket támasztanak. A kérdés csak az, hogy mekkora a kritikus tömeg, azaz hány magyar felhasználó szükséges ahhoz, hogy a magyarítás anyagilag megérje. Egyre több program esetében érjük el ezt a szintet. De van a maiaknál sokkal jobb megoldás is!

Egy program többféleképpen lehet magyar. Ha az első programsortól kezdve magyarok fejlesztik magyarok számára, akkor minden szövegrész magyarul és magyarosan lesz írva, nem kell másokhoz sem igazodni. Persze ez egyúttal az anarchia forrása is, mert a programok kezelői felületén még nem alakult ki egységes magyar elnevezés- és elhelyezésrendszer, amelyet mindenki elfogadna és alkalmazna. Több évig tartó vita folyt már arról is, hogy az egyik központi kategória magyarul file, fájl, állomány vagy valami más elnevezéssel szerepeljen. Mikor jutnánk akkor az összes szakkifejezés végére? Sokkal gyorsaban születnek és kerülnek használatba spontán módon az új fogalmak, semhogy idejében megegyezhetnénk, hogy mi legyen a magyar megfelelőjük.

A „magyarrá válás" másik elterjedt módszere, hogy világszerte terjesztett programokat a hazai szakemberek honosítanak, a programban szereplő szövegeket „magyar módra" helyettesítik, így mind a menük, mind a help (súgó), mind a hibaüzenet magyarul jelenik meg. Ez gyakran azzal jár, hogy a menüpontokkal együtt a gyorsbillentyűket is magyarítják, ami az angol verziót elég jól ismerő felhasználót pillanatok alatt kezdővé degradálhatja. (Hasonló a helyzet, amikor az angol billentyűzetet magyarra cserélik.)

Leginkább csak közkincs jellegű (shareware, freeware stb.) programoknál szokás, hogy egy külső fájlban megtalálható a help és a hibaüzenetek szövege, azokat a ráérősebbek kedvük és nyelvtudásuk szerint lefordítják, majd mások számára magyar nyelvterületen közkinccsé teszik. Ilyenkor a program kezelése és külseje nem változik, és a felhasználók szótárlapozgatás nélkül is el tudnak a programon igazodni. Egyes esetekben a teljes kezelőfelület is átkonfigurálható saját nyelvünkre.

Nyelvi virtuóz kerestetik!

A programok túlnyomó többsége a fent leírt egyik csoportba sem tartozik, és a felhasználók megszokják, hogy melyik angol kifejezésnek mi a jelentése a használat szempontjából. Ez sok esetben elegendő is, főleg ha jól működő, hasznos programokról van szó.

A honosítással szemben nekem vannak fenntartásaim. Mit ér egy program „magyarsága", ha annak állandóságával, stabilitásával baj van... Ha profitérdekből évente ki kell hozni a program újabb változatát, amely még többet tud, még jobban kiszolgálja a felhasználóját. Legalábbis látszólag. A felhasználóbarát programokat nagyjából a menük megjelenésétől kezdve számíthatjuk, előzőleg a „könnyedén megjegyezhető" 2-300 parancsot billentyűkombinációkkal vagy begépeléssel érhettük el.

A menüpontok kiválasztását hivatott felgyorsítani az a lehetőség, hogy a menüpontokban egy-egy betű ki van emelve, és annak a billentyűnek a leütésével aktivizálható az adott parancs. Ehhez már az eredeti angol nyelvű programokban sem volt elegendő az első betű (lásd Save és save As), és még nehezebben oldható meg a nyelvi változatokban. Az angol nyelvű fejlesztéskor is igyekeztek úgy elnevezni minden menüpontot, hogy legyen benne egy olyan betű, amely még szabad. Ahhoz gyakran nyelvi lelemény szükséges, hogy ezt az elvet a fordításkor is követni tudjuk. A fordítók rémálma pedig a változó menü, amelyben a menüpontok a program futása közben átalakulnak, és úgy kell megválasztani a megfelelő betűket, hogy egyetlen almenü-kombináció esetén se legyen ütközés.

A programok magyarításakor természetesen kötődni kell az eredeti szöveghez (elvégre ez mégiscsak fordítás), és az előbb részletezett finomságokat is



figyelembe kell venni. Ezen feltételek mindegyikét kielégítő nyelvi virtuozitás mellett még a számítástechnikát is elég jól kell ismerni, különben nem lehet értelmesen fordítani. Nem sok ilyen fordító lehet. Annak alapján is így gondolom, hogy nekem nemegyszer kellett egy program magyar nyelvű változatából kilépnem, és elindítanom az angol variánst, hogy megtudjam, mit is jelent egy bizonyos menüpont.

A nyelvi variánsok közötti kompatibilitás a Microsoft programjainak egyik gyenge pontja. Nem tanácsos angol nyelvű Windowson magyar nyelvű Wordöt futtatni, és viszont. Mivel az angol–angol kombinációnak van a legnagyobb felhasználói (és béta-tesztelői) tábora, itt derülnek ki leghamarabb a hibák, és emiatt legalább ennek mindig stabilabbnak kellene lennie, de sajnos ez sem mindig igaz. Továbbá léteznek mindenféle közprogramok Windows alá, amelyeket sokan használnak. A windowsos közprogramok szerzői többnyire szintén angol nyelvű környezetet használnak, ezért programjaik biztosabban futnak ezen, mint egy nemzeti verzión. Arról már nem is szólva, hogy ha a közprogram esetleg makrónyelven íródott, amelyet szintén honosítottak, akkor a programíró nyelvének meg kell egyeznie a mienkkel, hogy a programot használni tudjuk.

Volt szerencsém a Windows holland nyelvű változatához is. Ott valahogy még tudtam emlékezetből dolgozni, mert a menüpontok elhelyezkedéséből következtettem arra, hogy melyik mit jelenthet. Viszont amikor egy holland DOS hibaüzenetet kaptam, végképp megálltam. Szerencsére (vagy éppen ellenkezőleg) ilyen problémája kevés embernek akad.

Pont és vessző

Habár eddig az eredeti angol (amerikai) nyelvű programok használata mellett érveltem, tudom, hogy nem mindenütt megoldás a kizárólagos használatuk. Vannak nyelvi sajátosságaink, amelyekről nem szabad lemondani. Például kettős hosszú ékezetes betűinkről. A kalapos és hullámos o-k és u-k továbbra is ott hemzsegnek a HTML-szabványos honlapokon, sőt egyes falragaszokon is! Nekem az is sokáig tartott, amíg megszoktam, hogy nincs alsó idézőjel a billentyűzeten. A számítógép mellett ma felnövő ifjúságot pedig talán már arról kell meggyőzni, hogy a magyar írásban viszont van.

Sajátos feladat a határozott névelők kezelése a hivatkozásokban. Amikor azt írjuk, hogy „az ötödik fejezetben", majd utána átrendezzük fejezeteinket, akkor „a hatodik fejezetben" szöveget kellene írni. A hivatkozásokat egy terjedelmes könyvben sziszifuszi munka mindig átírni, inkább a programot kell felkészíteni rá, hogy tudja, mikor kell „a" és mikor kell „az" névelő a megfelelő hivatkozás elé. A megoldást nyilván csak saját fejlesztésű programoktól várhatjuk. Hasonlóképpen a magyar szavak elválasztását és a szavak szinonimáinak felsorolását sem várhatjuk el a nem honosított programoktól.

Számomra több mint másfél évtizeden át nyilvánvaló volt, hogy a számítógépeken és a számológépeken nem tizedesvessző, hanem tizedespont van. Bármilyen programmal találkoztam, ez



alól nem volt kivétel. Azután lett. Nem tudom, még hányan estek pofára ilyen programmal, én igen. Végeláthatatlan vitát lehetne folytatni arról, hogy a számítástechnikai hagyományokat, vagy a magyar nyelvben kialakult hagyományokat kövessük-e. Azt hiszem, hogy elvi fejtegetések helyett itt alapvetően pragmatikus szempontokról van szó, azokat kellene érvényesíteni, ha már kialakult ilyen abszurd „világhelyzet". A számítástechnika alkalmazásának tekintélyes részét teszi ki a szövegszerkesztés és a táblázatkezelés, tele pontokkal és vesszőkkel. Új játékszabályokat bevezetve igen nagy káoszt lehet kavarni. („Ha egy villamos új utakon akar járni, óhatatlanul kisiklik." — Stanislaw Jerzy Lec aforizmája.)

Amit mindenki megért

Nagyon sokan programoznak, és a több száz, sőt több ezer programnyelv közül alig van egy-kettő, amely magyar nyelvű lenne. A nyolcvanas évek közepén a Homelab számítógépre készült egy magyarosított Basic, amelyben a kulcsszavak magyar megfelelőikre lettek lecserélve. Már akkor sem tudtam elképzelni, hogy egytucatnyi angol szó lehetetlenné tette volna, vagy komolyan gátolhatta a programozás megtanulását. Azt esetleg még elfogadom, hogy a Logo nyelvet magyarítják (mint például a Szegeden készített Lógó rendszerben), mert a programozópalánták (azaz a megcélzott felhasználók) írni-olvasni is alig tudnak. Persze ezek is tört magyarságúak, a „Fordulj balra húsz fokkal!" egyszerűen „balra 20" lesz, míg a „Tollat fel!" helyett „nemír". Noha a Logo teljes értékű programozási nyelv lehetne, az óvodától az egyetemig elkísérve a tanulót, szinte mindenki leragad a látványos részeknél, így csak a teknősgrafika implementálásának van meg a magyar megfelelője.

Véleményem szerint a programozás nyelve továbbra is az angol lesz, angol nyelvű kulcsszavakat fognak tartalmazni a programnyelvek. Ez korántsem jelenti azt, hogy ne legyenek magyar szakszavaink. Ha életképesek, legyenek. Nem tudom, hogy az iteráció, a ciklus, a rekurzió vagy az objektum helyett található-e pontos magyaros kifejezés. A nyelvünkbe jól beilleszkedő idegen szavak mindig is meghonosodtak, nem kell mindenáron erőltetni helyettesítésüket.

Az APL, illetve a J korai verziója nem tartalmaz kulcsszavakat, pontosabban nem angol nyelvűeket. Helyette speciális karakterek vagy azok sorozata használatos. Mivel ezek a jelek a matematikai jelölésrendszerből származnak, egységesek, és viszonylag könnyen érthetők. Persze a fölöttébb tömör jelölésmódnak köszönhetően joggal illetik a „csak írható nyelv" jelzővel. Pedig ebből az irányból indult el a honosítás vagy nem honosítás dilemmáját feloldó, az elmúlt évtized során a gyakorlatban is jól bevált megoldás: az ikonosítás. Az ikon az a jel, amelyet minden felhasználó megért, függetlenül attól, hogy mi az anyanyelve. Ha a programírókban majd lesz olyan törekvés, hogy a világ soknyelvű, soknemzetiségű lakosságának egyenlő esélyeket adjanak, hogy minden országban megkönnyítsék a számítástechnika alkalmazását, hogy gyorsítsák az elmaradott területek informatikai felzárkózását is, akkor a programok futtatható bináris részében csak nyelvfüggetlen jeleket és ikonokat fognak elhelyezni, hogy a nyelvi modulok (menüszövegek, leírások, helpek, hibaüzenetek stb.) kívülről, nagyrészt bárki által könnyen módosítható formában kapcsolódhassanak a rendszerhez. Ez mindenki előtt megnyitná a saját nyelven történő használat lehetőségét, megszüntetné a verziók közötti ütközések egyik okát, és a bevezetőben említett kritikus tömeg probléma egyszeriben megoldódna.

Aszalós László
aszalos@godel.math.klte.hu

Mit üzen a képernyő?

A Microsoft programjainak lokalizációja

Amikor a lokalizációt hallom emlegetni, fülembe cseng a gyerekkoromban hallott sláger: „Tengerparton áll egy kis lokál...” Persze, eszembe juthatna a lokális változó is a programozás órákról, de annak kevésbé emlékezetes a dallama. Miért is kell ez a szó? Miért fogadunk el egy újabb jövevényt? Miért nem fordításról, magyarításról beszélünk?

A tengerparti, kikötői lokálok, vendéglők sajátossága a nemzetközi közönség. „Négy különböző nemzetiség képviselője volt az asztalnál: egy amerikai gyalogos, egy francia őrvezető, egy angol géppuskás és egy orosz hússaláta. A gyalogos, az őrvezető és a géppuskás a padon foglaltak helyet, a hússaláta az asztalon, egy tálban” — olvashatjuk a klasszikus regénykezdetet (Rejtő Jenő: Három testőr Afrikában).

Amikor a programok honosításáról beszélünk, négynél is több nemzetiség képviselőire kell gondolnunk, ami nem vág egybe a „hon” szóval. Nemzeti nyelvi változatok létrehozásáról van szó, a programok helyivé tételéről. Ez a lokalizáció. Talán jobb lenne a néha szintén előforduló nemzetiesítés, de a latinos alakokat általában elviseljük, és legfeljebb dúdolgatás közben gondolok arra, hogy valami mást kellene találni.

Nemzeti változatok

Alakulgat a magyar Windows, a magyar Novell, sőt lassan a magyar Linux is. Meg a cseh, a svéd, a kínai... A helyi. A nemzeti. A lokális. A lokalizált. Ennek nem mindenki örül: vannak, akik szerint az igazi csak az eredeti, az angol változat lehet. Miért is lenne rossz a honosított program, alkalmazás? Erre két okot szoktak felhozni:

1. Mert a felhasználó is megérti.
2. Mert rosszul van honosítva.

1) Miért baj az, ha a felhasználó megérti, mi van a programban? A számítástechnika sajátos tudásmonopóliumot hozott létre. Azok, akik értették (vagy úgy tettek, mintha értenék), tiszteletet váltottak (váltanak?) ki a laikusokból. Az ilyesfajta elismerés aprópénzre váltható. A tudásmonopóliumnak van egy valóságos része, annak a programozónak a tudása, aki egy bonyolult rendszert létrehoz. Persze, neki is dokumentálnia kellene a művét, érthető formában, és ha nem teszi, ez nem tisztességes tőle. Mindenesetre információk visszatartásával biztosíthatja nélkülözhetetlenségét, ami megfelel a „struggle for life” (küzdelem a létért) törvényének. A tömeggyártók azonban nem érdekeltek az egyéni programozók ilyetén monopóliumának fenntartásában, ellenkezőleg, nekik az a jó, ha a felhasználó egy kis segítséggel, általános műveltsége felhasználásával önmaga is alkalmazásba tudja venni a programot.

2) Hogy a honosítás mennyire sikerül, az viszonylagos. Meglehetősen sok feltételnek kell eleget tenni ahhoz, hogy egy lokalizált változatot ki lehessen adni, illetve a kritikusok ne találjanak benne kivetnivalót. A lokalizációval kapcsolatosan sok a technikai vita, ezeket most kikerülöm, itt, a lokál gyalult asztalánál csak az eredmény a fontos.

A gyakorlat azt mutatja, hogy a nemzeti változatok a felhasználók körében lényegesen népszerűbbek az angolnál. Kérdés, hogy a gyártó cégek hogyan élnek, és hogyan élnek vissza ezzel a népszerűséggel: jó szolgálatot tesz-e a lokalizáció, vagy ellenkezőleg, nyelvetszakmát rongál. Az utóbbiakhoz szeretnék a továbbiakban néhány gondolatot hozzáfűzni.

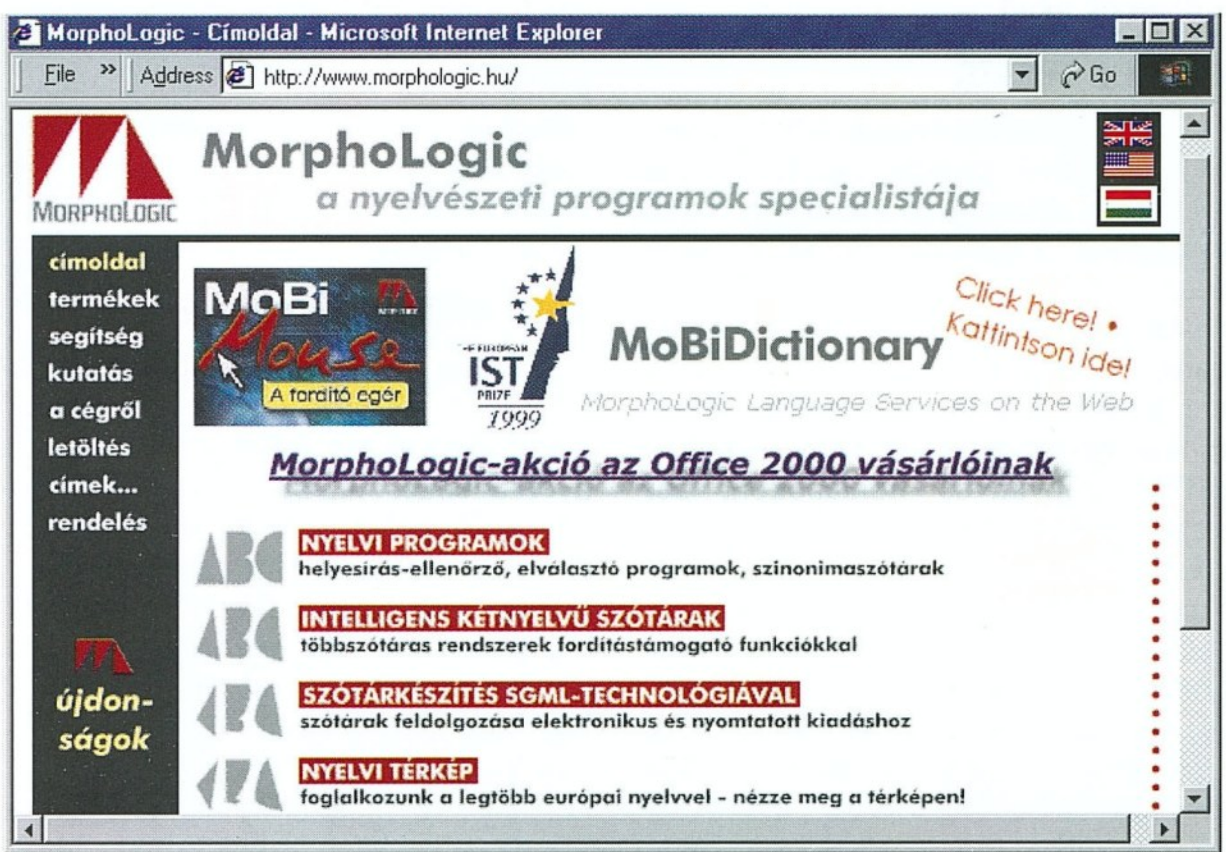
Menüdilemmák

Régi Microsoft-felhasználó vagyok. Az élet úgy hozta, hogy a Windows és a Word révén ehettem meg kenyerem javát. A Microsoft termékek lokalizált, magyar változatáról van tapasztalatom. A DOS-os, TeX-es világ után a grafikus felületen Windows 3.1-gyel és Word 2-vel kezdtem, ma pedig a Windows 2000 és a Word 2000 a munkatársam.

A lokalizáció a közhit szerint az alkalmazás használata során megjelenő, láthatóvá váló szövegek átírását jelenti. Bár a honosítás célja ennél összetettebb, kétségtelenül ez a leglátványosabb része.

Elsőként az alapképernyőn megjelenő szövegeket, főképp magyar nyelvű megjelenésüket, illetve következetes használatukat veszem szemügyre. Két fajtájuk van: a menüben megjelenő elnevezések, és az ikonokhoz írt magyarázatok, amelyek akkor jelennek meg, ha az egeret az ikonra közelítjük.

A menük elnevezései hagyományosnak tekinthetők. A közönség fokozatosan megszokta őket, már nem vitatják úton-útfélen — bár éppenséggel lehetne —, hogy miért fájl a Fájl. A számítógép-specifikus terminológia elég kiforrott. Volt rá ideje. Bármennyit fortyogott is azonban, a fájlt nem tudja kilökni



magából. Nem titkolom, nekem nem szimpatikus ez a szó. Nem lehet kiejteni. És ráadásul nem is olyan fontos. Nevezhetnénk bárminek. Például, a Wordben dokumentumokat (és sablonokat) készítünk, mentünk, nyitunk meg, az Excelben táblázatokat vagy munkalapokat stb., és nem fájlokat. Ha a Fájl menüt legördítjük, érdekes módon csak dokumentumra és weblapra van hivatkozás, a fájl szó többé nem fordul elő. Mégis, a fájl kell a magyarnak! Nem is számítástechnikai szöveg az, amelyben nem fordul elő legalább 3 és félszer.

A Szerkesztés, Nézet, Beszúrás, Formátum, Eszközök, Táblázat, Ablak menüelnevezések könnyen polgárjogot nyertek. Ezek sikeres tükörfordítások, egy részük nyilvánvalóan megfelel az eredetinek. A Nézet ugyan nem hétköznapi módon felel meg a View-nak, a szótár alapján mások talán nem ezt a szót választották volna, de megértése nem okoz gondot. Ugyancsak tökéletesen érthető a Formátum, bár itt az eredeti Format alak latinos változatát választotta az egykori honosító. Ez tartalmilag nem teljesen helyes, hiszen ez a menüben a Szerkesztés párja, a helyes neve Formázás lenne, annál is inkább, mert azokat a jellemzőket, amelyeket a magyarban összességében formátumnak nevezünk, nem is itt állítjuk be, hanem a Fájl menü Oldalbeállítás parancsával.

A Windows menüsorának sokat vitatott eleme a Súgó. A Word korábbi változataiban alkalmazott ? helyett ma már kiírják ezt a szót. Mi is a baj vele? Mondják, hogy nem illik egy komoly dologban a diáknyelvet alkalmazni. Érdekes, régebben én is azt hittem, hogy ez teljesen iskolai eredetű, mondtam is, hogy akkor már a puska jobb lenne. Azután megvilágosodott: a súgó színházi szakszó is. Ugyancsak komoly emberek mégis azt állítják, hogy a „help” segítő vagy segítség, nem pedig súgó. Vajon megengedheti-e magának a lokalizátor, hogy eltávolodjon az eredeti terminológiától, és ne ugyanazt adja vissza?

Az amerikai terminológiai szokások különböznek az európaiaktól, ami okoz némi gondot, azonban lehet hozzá alkalmazkodni. Főleg az okoz nehézséget, ha az amerikai elnevezés rossz, semmitmondó. Vegyük például az Options esetét. Ez a szó nehezen magyarítható. A jogi nyelvben előfordul az opció, de azt sehogy sem lehet a mi konkrét eseteinkre értelmezni. Kezdetben Egyebeknek fordították, amiről ugye, semmit nem lehet tudni a neve alapján, inkább csak az Izé eufemisztikus változata. Mára Beállítások lett belőle, és ez önmagában is utal a tartalmára. Visszatérve a Súgóra. Ebben az esetben szerintem nem kellett volna eltérni a szó szerinti fordításától, a változtatás azonban élénkítette a terminológiát, és bizonyos értelemben feloldja a nem számítástechnikus felhasználók idegenkedését. Ez a szempont a túl szigorú európai terminologizálás feltételei között nem mellékes.

Magyaros vagy csak lokális?

A lokalizáció értékelésekor azzal is foglalkozhatunk, mennyire magyaros az eredmény. Átnézve a Word 2000 menürendszerét, az alábbi idegen elemeket találtam:

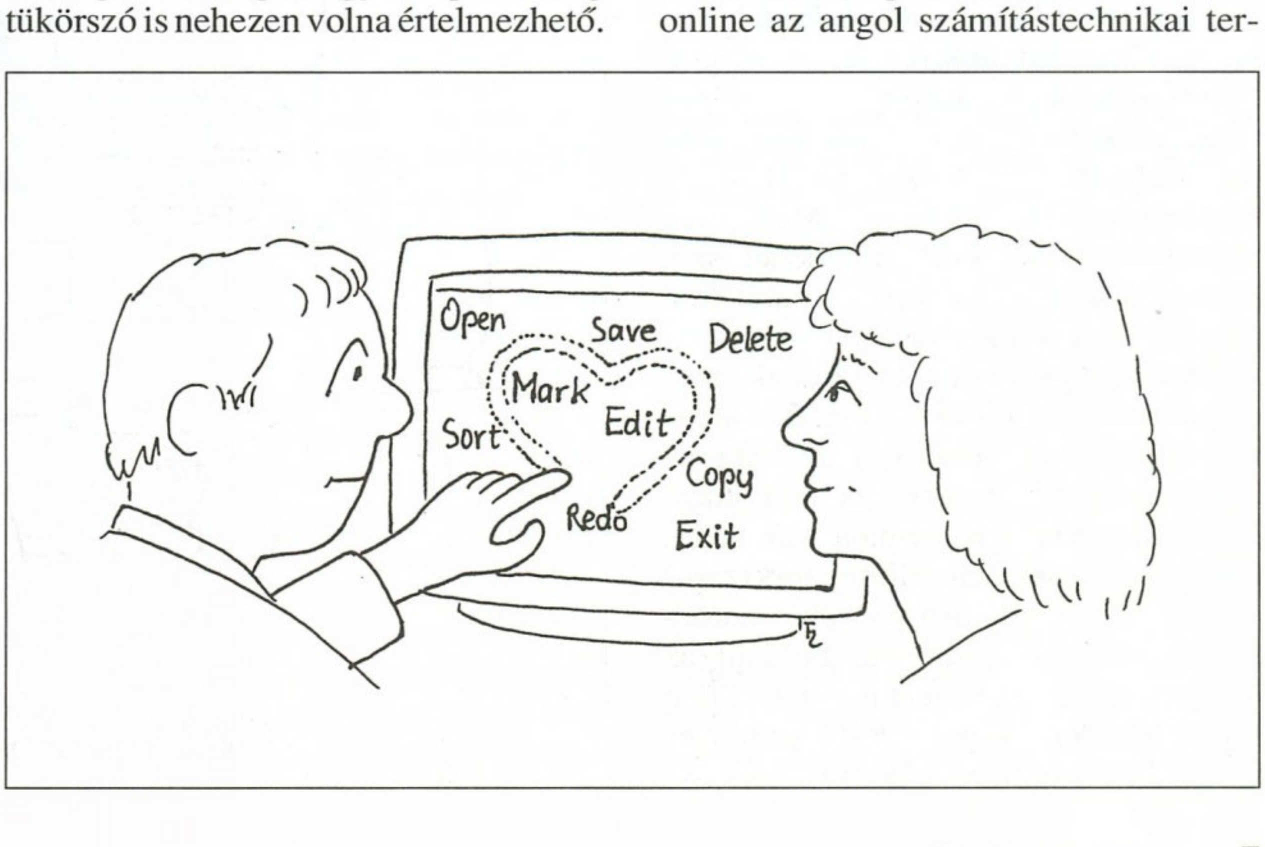
■ A Fájl menüben maga a fájl, de ez csak a menü nevében fordul elő. A parancsnevek között a weblapot találjuk (a dokumentum már nem nevezhető idegen szónak). Mélyebbre hatolva sem találunk újabb elemeket, említhető a PostScript és a letter a Nyomtatás panel Egyebek alpaneljén, azonban az előző márkanévként nem változtatható, a letter pedig végső soron egy mértékegység neve, amellyel azért nem érdemes foglalkozni (bár honosíthatnánk levélnek), mert a jelölt fogalmat (21,59x27,94 cm-es papírformátum) mi nem használjuk (mellesleg az Oldalbeállítás panelon a papírméretek között más idegen méretnév is található).

■ A Szerkesztés menüben jön elő a hiperhivatkozás mint neologizmus; nem teljesen újdonság, mert már a Word97 is tartalmazta. Hogy ez a szó mennyire tekinthető idegenszerűnek, az vitatható. Figyelemre méltó azonban a képzési módja: ez a hypertext magyarítása. A hiper szó jelentése magyarul nem igazán világos, így a hiperszöveg tükörszó is nehezen volna értelmezhető. A hiperhivatkozás éppenséggel alkalmas erre, mert bár a hiper továbbra sem tisztán érthető, a jelenség a hálózati terminológiában közismert hivatkozások egyike, s mivel állományok közötti mozgást tesz lehetővé, a túlzó, magas fokra utaló hiper összetételi tag előképzőszerű elem, és hozzásegít a megértéshez.

■ A Nézet menü a Webes elrendezés parancsot kínálja. A lokalizátorok a web szót lényegében magyarnak tekintik. A web az egész világot átfogó egyik hálózati forma, de az itt használt alak nincs összefüggésben a hálózattal, csak az ott alkalmazott szövegformázási móddal. Ezért a Webes elrendezés kifejezés kifogásolható, és remélhető, hogy ezen a területen is kialakul majd olyan terminológia, amely az összefüggéseket következetesen tükrözi. Mindazonáltal a kifejezés érthető, betöltheti a szerepét.

■ A Formátum menü terminológiai tekintetben nem mutat újdonságot a korábbi változatokhoz képest, konzerválja azonban a korábbi lokalizációs folyamatban keletkezett rossz elnevezést. A Bekezdés panel Szövegbeosztás fülén találjuk A fattyú- és árvasorok nevű választónégyzetet. Ez az angol Orphan and Widow pontatlan fordítása. A pontatlanság minősített konfúziót takar: a magyarban a fogalmat kizárólag fattyúsornak hívják, mert nálunk nem jelenik meg az a kettősség, amelyet az angol terminológia tükröz, hogy a hasáb tetejére került kimenetsorról (özvegy), vagy a hasáb aljára került kezdősorról (árva) van-e szó. A lokalizáló ezért melléfogott: az *özvegy* és az *árva* párossal még ki lehetett volna fejezni a különbséget, de a *fattyú* és az *árva* így együtt teljesen zavaros.

■ Az Eszközök menü az Online közös munka paranccsal tűnik ki. Az online az angol számítástechnikai ter-



minológia igen nehezen átültethető szava. Ez részint a sokszófajúságával függ össze, másrészt pedig széles értelmezési lehetőségeivel. Mivel a konkrét értelemben feltételezhetően arról van szó, hogy a közös munka a hálózatra egyidejűleg rákapcsolódott munkatársak között folyik, talán megoldás lett volna a „hálózati közös munka" kifejezés. Nézetem szerint az „online közös munka" többet sugall, mint ami annak tényleges tartalma.

A párbeszédablakok megőrizték a már klasszikusnak számító OK parancsgombot, mellette szerepel a sok vitát kiváltott Mégse. Ismeretes, hogy a korábbi változatokban a Mégsem szó állt a parancsgombon, ez rövidült le a Windows 98 tájékán. A magyarázat erre az lehet, hogy amikor a gombok segítségével kommunikálunk a géppel, akkor ez a gomb egy ilyesféle mondatot helyettesíthet: „Mégse növeld a nagyítást!" Az Interneten elég éles vita folyt erről, amelyből világosan lehet következtetni arra, hogy a megszokás nagy úr — sokan hiányolták a megszokott Mégsemet.

Menük és ikonok

A lokalizációval szemben lényeges minőségi követelmény, hogy a menüparancsok és a hasonló funkciót betöltő ikonok elnevezése megegyezzen. Példák a Word 2000-ből:

Ikonmagyarázat	Menüparancs
Nyelvhelyességi ellenőrzés	Nyelvi ellenőrzés
Hiperhivatkozás beszúrása	Hiperhivatkozás
Oszlopok	Hasábok
Balra igazítás	Balra zárt

Ezek közül bizony súlyos lokalizációs hiba a hasábok és az oszlopok összekeverése. Ez a Word 2000 újdonsága, és különösen kellemetlen, hogy az Oszlopok szó használata az adott fogalomra teljesen hibás. Még csak terminológiai zavarról sem lehet szó, mint a Word változataiban a keret és a szegély szavak esetében.

Hiba, hogy a sorok igazításának megnevezései között a Balra igazítás kivétel, középre és jobbra már zárja az ikonnév a sort. (A helyes megoldás: Igazítás, melynek 4 formája van, balra, középre és jobbra zárt, illetve sorkizárt.)

Tanulságos a Külső szegély eszköztár neveinek lokalizálása. Például az összes lehetséges esetet tartalmazó ikon neve: Minden szegély. Elég nehézkes megoldás, értelmezni ekképp lehet: „Az összes szegély". Tulajdonképpen önmagában a „belső szegély" is kicsit különös, de a „Lejtő átló" is szép talányos lelemény.

A szövegátírás régen ismert problémája, hogy a rendelkezésre álló helyet az eredeti nyelvű program határozza meg. A Word 2000 ebben a vonatkozásban határozottan hátrányos. Tipikusnak mondható, hogy a lokalizáció eredményeként létrejövő magyar nyelvű panelokban hiányosak a feliratok. Például az összes fájlkezelő parancsban ez szerepel: Dokumentu..., A Szerkesztés menü Keresés és csere paneljában ilyenek vannak:

```
Kis- és nagybetű megkülönbözt
Csak ha ez a teljes s
Behelyettesítéss
```

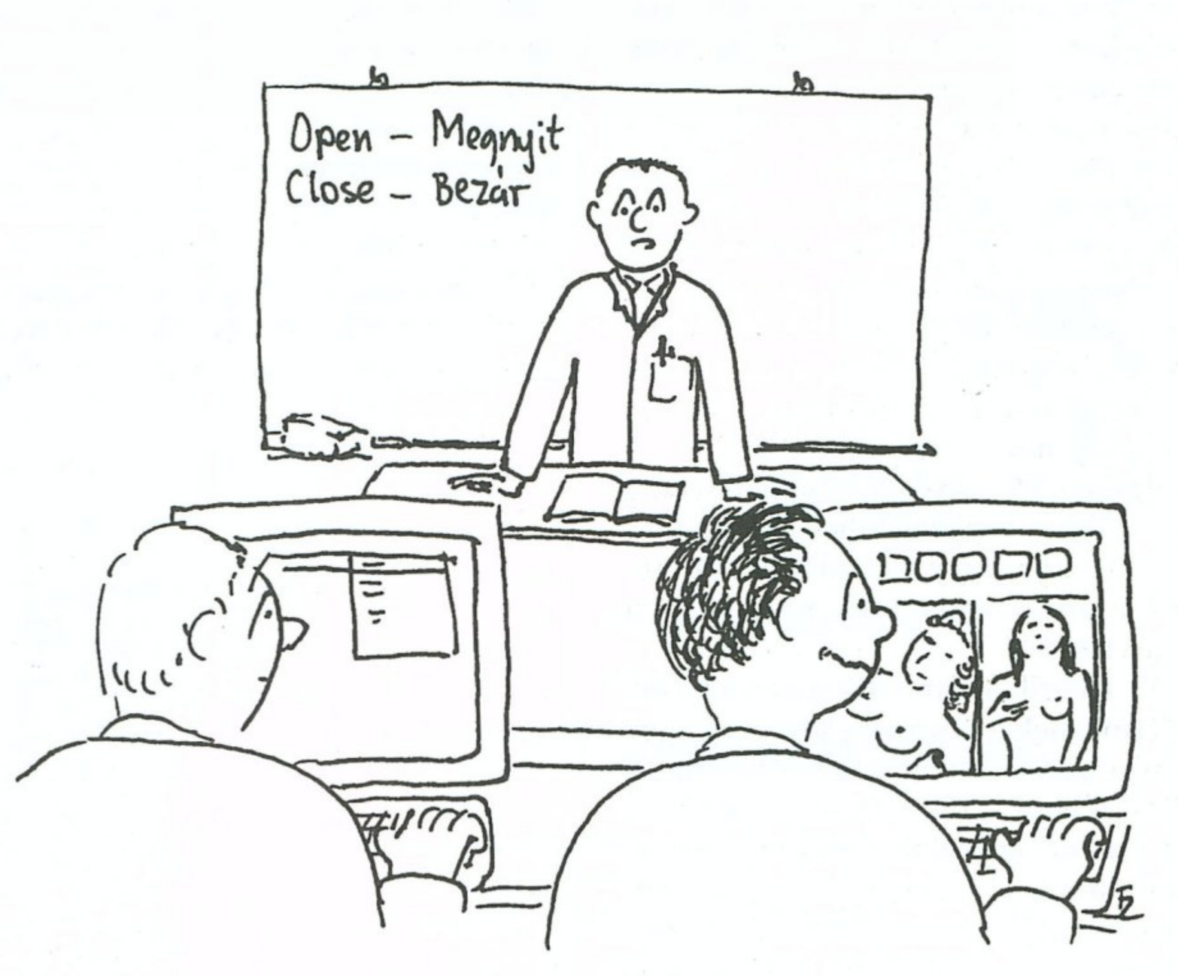
Ez utóbbi a Behelyettesítéssel szó maradványa, és folytathatnánk a hibás panelok felsorolását, de felesleges, mert mindegyik jellemzője a csonkulás, és annak oka is mindig ugyanaz.

Amikor a lokalizációról vitatkozunk, ma már túlléphetünk az egykori alapkérdésen: legyen-e magyar nyelvű alkalmazás, vagy ne legyen. Azok a korábbi elképzelések, hogy egyszerűbb nyelvet tanulni-tanítani, mint fordítani-lokalizálni, az élet más területein is tévesnek bizonyultak. Az Európai Unió hatalmas tömegű nyelvi munkát ad a kifejezetten erre a célra kiképzett fordítóknak és tolmácsoknak, s mi több, az utóbbi időben olyan törekvések is vannak, hogy mindenki csak az anyanyelvére fordíthasson.

Az idősebb számítógépes generáció szemtanúja lehetett annak, hogyan tanult meg a számítógép írni-olvasni. Most azt figyelhetjük meg, hogyan képes elsajátítani az idegen nyelveket. A lokalizáció értékelése most már a minőség kérdését veti fel. Aki régóta használja a magyar nyelvű termékeket, határozott fejlődést figyelhet meg. Fejlődik a koncepció, a lokalizáció vezetői tekintettel vannak arra, milyen hatást képesek gyakorolni a nyelv egészére. Nem mondhatjuk, hogy már minden tökéletes, hiszen a fentiekben kritizált hibákat egyetlen termékből idéztem, és annak sem teljes körű áttekintése alapján.

A lokalizáció történetében váltást vehettünk észre a Windows 95 megjelenése körül. A váltás pozitívnak ítélhető meg, de újból elindított egy kristályosodási folyamatot. Talán célszerű lenne, ha a lokalizálók kialakítanának valami egyszerű mechanizmust, hogy a felhasználók gyorsan és könnyen jelezhessék észrevételeiket. Olyan bonyolult rendszer esetében, mint a Word szövegszerkesztő vagy más hasonló alkalmazás, a hibák teljes kiküszöbölése nem egyszerű feladat, és a felhasználók ebben bizonyára szívesen részt vennének.

Kis Ádám



— A működtetés filozófiáját próbáljátok megérteni, a kezelőszervek pontos kialakításától függetlenül.

Magyarítás

Fordítani csak pontosan, szépen...

Magyarítás... Nagyon jó kifejezés. Bárki ötlötte is ki, megfogta a lényeget. Érezhető belőle, hogy itt valami művi, „csinált" dologról van szó, nem természetes folyamatról...

Ha egy irodalmi alkotást — regényt, verset, dalszöveget — akar valaki egyik nyelvről a másikra átültetni, műfordításról beszélünk. Mindenki tudja: a cél nem a szavak, kifejezések pontos átvitele egyik nyelvről a másikra, hanem a gondolatok és érzelmek átadása a mű olvasójának, hallgatójának. A zseniális műfordítók nem a „szótár" nagy ismerői, hanem az adott nyelvet természetes közegként (anyanyelvként) használók gondolatvilágát ismerő, vagy abba a világba önmagukat jól beleélő emberek. Nem lehet véletlen, hogy az igazi műfordítók rendkívül ritkák, illetve hogy aki egy művet eredetiben is képes élvezni, annak irodalmi értékeit átérezni, az már nagyon magas fokon ismeri a nyelvet.

„Szakfordítás"

A szakirodalom fordítását szinte sosem nevezzük műfordításnak. Itt csak sima „átültetésről" lehet szó. Ebben a kifejezések jelentésének minél hívebb továbbítása a cél. Ha a forrásnyelv az adott szakma alapnyelve, nem is fordítanak le mindent. Akinek a szakkönyv szól, az a szakszavak egy részét eredeti nyelven is érti, sőt gyakran csak azt érti meg, és a fordítással nem tud mit kezdeni. (Erről majd még lesz szó.) A jó szakfordítónak nem annyira a forrás nyelvét, mint inkább a szakmát magát kell nagyon alaposan ismernie.

A számítógépek világában elterjedt fogalom a magyarítás. Akkor beszélünk erről, ha tudjuk: az adott program, környezet kifejezéseinek fordításakor ezernyi kompromisszumot kell kötni, akár a felhasználható szavak hosszának korlátja, akár az eltérő ragozás vagy más szempont miatt. (Az angol elöljárókat, a magyar ragokat használ, ebből szép kis mondatszörnyek keletkeznek, ha egy program az üzeneteit önmaga „szerkeszti" össze.) Érthető tehát, hogy időnként a legnagyobb jóindulattal is csak valami öszvér megoldás születik, amelyet egyik fél sem tud teljes értékűnek elfogadni. A hozzáértőt idegesíti, esetleg nevetésre készteti, a kezdőt elijeszti, a nyelvőrt pedig szent harcra ingerli.

Miért magyarítunk bizonyos dolgokat, másokat meg miért nem? Érdekes probléma. Mindenki szeretné, ha az elérhető programokkal minél kevesebbet kellene bajlódni. Könnyű legyen megtanulni, használni. Ennek egyik gátja lehet a nyelvismeret hiánya. Nem várható el mindenkitől, hogy megtanulja azt a fura képződményt, amit angol számítógépes szaknyelvnek nevezünk. Még a magyar szaknyelvet sem könnyű elsajátítani (ha egyáltalán van ilyen). Jogos igény tehát, hogy amit lehet és kell, azt fordítsuk le, hogy az is dolgozhasson vele, aki nem értené meg másképp. De mit és milyen mélységig szabad lefordítani? Eszem ágában sincs ezt a vitát eldönteni, rábízom a hivatásos és amatőr nyelvészekre, csak szeretném néhány vadhajtásra felhívni a figyelmet.

Hány „eredeti" magyar szó van?

Ha egy nép gyermeke kitalál egy azelőtt sosem létezett dolgot, új fogalmakat is alkot. Meg kell nevezni a tárgyat, a használatához szükséges teendőket, és mindazt, ami az eszköz alkalmazásához kapcsolódik. Ezt megteheti úgy is, hogy új szavakat alkot (eredetieket kreálva, vagy a régiek összekapcsolásával, átszabásával...), de úgy is, hogy a meglévőkre rak rá újabb jelentést. Ezeket azután a környezet vagy elfogadja, vagy elveti. Bárhogy is végződik, az új fogalmak az adott nyelv részévé válnak. Ha e fogalomkörrel más nyelvet beszélők is szembesülnek, számukra két választási lehetőség van: vagy átveszik az eredeti nyelvből a fogalmat kifejező szavakat, vagy újakat alkotnak erre a célra, saját nyelvükön. Mindkét megoldásra ezernyi példa van. Nem lévén nyelvtörténész, csak sejtésem van arról, hogy például e cikk eddigi szavai, kifejezései közül mennyi lehet az eredeti magyar, mennyi az átvett, mennyi az átvett és kissé eltorzított, az eredetire már alig hasonlító, mert évek során a magyar nyelv hangrendjéhez idomult.

A honfoglalás után, a kereszténység felvételekor őseink teljesen új, addig ismeretlen fogalmakkal találkoztak, amelyek megnevezését jórészt megtanulták az itt élt népektől. A földművelés ezernyi finom árnyalatának megnevezésére biztosan nem volt szókincse egy nomád, főleg állattartásból és harcból élő népnek. Azt át kellett venni a szomszédoktól. Ahogy a vallási élet új formájának nyelvezetét is. Később ennyire alapvető változások már nem történtek, de a törökök, a németek, az oroszok mind itt hagyták nyomaikat nyelvünkben.

Manapság rengeteg új fogalom, új találmány születik, amelyek kifejezéseit jobb híján át kell venni másoktól, majd ezeket átalakítani saját „képünkre". A számítástechnika is ilyen új technológia, és ahány ország, annyiféle módon próbálnak megküzdeni a befogadás ellentmondásos folyamatával. Egyik nemzet sem veszi át szívesen az idegen szavakat, inkább igyekszik sajátokat alkotni helyettük.

Természetesen nem jó, ha mindent mechanikusan átveszünk, és ahol lehet, magyar szavakat kell alkotni. De csak mértékkel és gondolkodva. Az egészséges nyelvérzék a nagyon művi szavakat úgyis kiveti magából. Ha pedig egy kifejezés meghonosodik, jön a másik nagy vitatéma, a helyes írásmód.

Nemrégiben az internet egyik levelezőlistáján valaki megjegyezte, hogy a ventillátor szó egy l-lel, ventilátorként írandó. Nosza jöttek a válaszok pró és kontra. Fellapozták az elmúlt évek helyesírási szabályzatait, és ha úgy tetszik, mindkét félnek igaza van. A régebbi szabályzatok szerint magyarul a két l-lel írt alak a jó, de jó egy évtizeddel ezelőtt megváltoztatták, és ma az egy l-es írásmód minősül szabályosnak. (Ki tudja miért, talán már az EU-konformitás jegyében, merthogy angolul is egy l-lel írják.) Nem szerencsés dolog arra kötelezni embereket, hogy 40-50 éven át gyakorolt helyesírásukat bármiféle nyomós érv nélkül megváltoztassák. Az Akadémia úgy döntött: a ventilátor a jó, az öregek meg úgy: a ventillátor. Senki sem harapja el miatta a másik torkát.

Zsargon és szaknyelv

A dolgok helyes megítéléséhez kell egy kis rugalmasság is. Attól, hogy

néhány száz vagy néhány ezer ifjú internetező semmibe veszi a helyesírás szabályait, mert az úgy „cool", még nem kell vészharangokat kongatni. Mindenféle dörgedelem nélkül is kinövik. Ezek a „cool guys", ha más közegben írnak, törekedni fognak rá, hogy mások is megértsék őket. A „szabad" írásban kialakult szavak, szófacsarások, hunglish kifejezések, ly-j felcserélések általában csak az adott környezetben használatosak, és akárcsak a diáknyelv vagy a jassznyelv szavai, ezek is jönnek és mennek, elmúlnak, és átadják helyüket más átmeneti képződményeknek.

A szaknyelv megítéléséhez is jó, ha okulunk egy kicsit a történelemből. Egy szakmát a magyar népesség 10 millió tagjából gyakorol mondjuk 50 ezer. Ha az adott szakma új, akkor azelőtt sosem volt szavakat használ, többnyire átvéve a forrásnyelvekből. Tetszik vagy sem, ezen az 50 ezer emberen múlik, hogy a magyar nyelvbe melyiket fogják beilleszteni, melyiket átalakítani, és melyik helyett találnak „eredeti" magyar megfelelőt. Ha ezek a szavak már széles körben elterjednek, akkor átesnek a köznyelvbe bekerülő szavakra jellemző alaki és jelentésbeli változásokon, 10–20 év múlva pedig vagy elfogadott szavak lesznek, vagy elfejtjük őket. Ma beszélt nyelvünk szavainak túlnyomó része így keletkezett. Ha mindig szigorúan ragaszkodnánk a magyar kifejezéshez, szegény zsiráfot (giraffe) foltos nyakorjánnak kellene neveznünk, és legfeljebb arról lehetne jó kis akadémiai szócsatákat vívni, hogy a nyakorján j-vel vagy ly-nal írva helyes.

Nem feltétlenül káros mindaz, ami ma idegenül hangzik. A szakma számára fontos az egyértelműség, és egy jó fordítás sem mindig tudja azonnal helyettesíteni a jövevényszót, néha jelentésbeli különbség is kialakul köztük. Aki ismeri a dűzni kifejezést, nem biztos, hogy a fúvókát is ugyanúgy érti. Az alapfokú szakkönyv akkor jó, ha minél kevesebb szaknyelvi kifejezést hagy feloldatlanul. Inkább legyen ott minden szó mellett annak magyarázata is, mintsem bizonytalan legyen az értelme. A magasabb szintű szakkönyv már nyugodtan használhat több idegen terminus technicust. Sőt, egy szakma első néhány évtizedében ez kifejezetten ajánlatos is. Ha már köznyelvivé válik a fúvóka, akkor elhagyható (és elhagyandó!) a dűzni. 30–40 évvel ezelőtt az öreg drehusok vagy autószerelők szakkifejezéseit nem nagyon értette meg egy kívülálló. Ma már legtöbb eszközükre megvannak a magyar kifejezések, így a „normális" ember sem tátog, mint a partra vetett hal, ha ilyen beszélgetésnek lesz fültanúja. Nem kényszerítette senki az áttérést a „magyarra". Néhány szó beépült a köznyelvbe, így mindenki számára érthetővé vált, másokat meg önként cseréltek ki a szakma gyakorlói, mert őket is zavarta az eredeti szó kifacsart változata. (A drehus — és a drehás is — egyébként esztergályos, ha valaki nem ismerné.)



— Tegnap próbáltam ismerkedni az emillel, de nem találtam a kukacot, így aztán a férjemmel brózoltunk egy jót a hálón...

Nyelvi gettó

Ma az informatika az a szakma, amelynek szemünk láttára, fülünk hallatára, sőt általunk kitalálva alakulnak a kifejezései. Nagyon fiatal. Néhány évtized múlva esetleg másik foglalkozás (például a biotechnikus) szakszavain fogunk hüledezni, a miénk meg addigra már természetessé válik.

A fordításnak egyébként is vannak határai. Értelmetlen lefordítani egy operációs rendszer parancsait, egy programnyelv utasításait, függvényeit. Napi gyakorlatként futunk bele abba az őrületbe, amit a Microsoft elkövetett: a Word makrónyelvének kifejezéseit és az Excel függvényeit „sikeresen" lefordították magyarra. Pontosabban valami magyarra emlékeztető szörnyűségre. Jól megmagyarították. Ennek eredményeként, ha egy magyar Excelből angol Excelbe kell áttenni egy táblázatot, a szülő és a gyermek nem értik egymást. Fordítva sem. Ráadásul a megszokott függvénynevek pontosan fedték jelentésüket, a magyarítás eredménye pedig sem magyarul, sem angolul nem jelent semmit. Sokszor csak úgy tudom értelmezni, ha előbb megpróbálom nyelvtudásom és egy szótár segítségével „visszaangolítani" a kifejezést, majd annak eredményéből deriválni, hogy mi volt az eredetije.

Ez a fajta magyarítás azért is káros, mert aki ilyen programmal dolgozik, nyelvi gettóba kerül. Csak a saját nyelvén kommunikáló programverziót tudja hatékonyan kezelni. Pedig bárkivel megeshet, hogy egyik percről a másikra idegen környezetben, másik cégnél vagy másik országban kell dolgoznia. Ha ehhez a sokkhoz még az is hozzájön, hogy megszokott eszközeit újra kell tanulnia, mert egy elhibázott koncepció miatt elferdítették az eredeti utasításokat és függvényeket, hát...

Ha egy program segítő szövegét (help vagy súgó), menüpontjait lefordítják magyarra, az még segítheti a betanulást (igazából a menük magyarítása már határeset, mert egy másik fordítás egy másik programban zavaró lehet). A hibaüzenetek helyes fordítása is segíthet jobban megoldani a problémát, de a függvények kifacsarása már agyrém. Nagyon göröngyös a terep. A programok nagy része például valami szótárszerűségből rakosgatja össze a hibaüzeneteket, így a már említett ragozás kontra elöljáró probléma nyelvileg torz üzeneteket eredményez.

Fordítani csak pontosan, szépen... és amit elkerülhetetlen, azt érdemes.

Sándor Gábor

Kötődés és elszakadás

Számítógépes nyelvünk a szakszótárakban

Bár a számítástechnikának voltak gyökerei német és orosz nyelvterületen is, fejlesztésébe az Egyesült Államok fektette a legtöbb pénzt, és alkalmazása is ott vált először tömegessé. Az informatika terminológiája a jelek szerint ugyanúgy szükségszerűen lett angol, ahogy az orvostudományé a latin. De a folytatás nem ugyanaz...

Az angol nyelv behatolása a többi nyelvbe konfliktushelyzeteket okozott, különösen ahol a nemzeti nyelv dominanciájának igénye mélyen gyökerezik a hagyományokban. Franciaországban például drasztikus intézkedéseket vezettek be az anyanyelv védelmére, a skandináv országok nyelvművelő intézményrendszert fejlesztettek ki, Hollandiában a világhírű Addison-Wesley szakkiadó leányvállalata holland nyelvű fordításokat és eredeti holland szakirodalmi műveket jelentet meg... Vannak viszont kis és közepes országok, amelyek nem is nagyon törekszenek a számítástechnika anyanyelvi szakirodalmának kialakítására, mondván, hogy nekik gazdaságosabb a szakembereket angolul megtanítani, mint lefordítani a műveket.

A számítástechnikai szakmában Magyarországon kezdettől fogva volt egy „angol párti" réteg, amely ugyan nem hirdette nézeteit harciasan, de nem is törődött azzal, hogy a szaknyelv magyar legyen. Felfogásuk jó nyelvtudáson alapult, így személy szerint nem volt szükségük fordításra, és sokan arisztokratikus elzárkózással igyekeztek saját szerepüket misztikussá tenni, annak minden előnyét kihasználva. Velük szemben álltak a „mindent minden áron magyarítani" akarók.

A gyakorlatban e kettő között kialakult viszonylagos egyensúlynak köszönhetően a hazai számítástechnikai szaknyelv korántsem annyira angolos, ahogy azt egyes nyelvművelők hangoztatják, ugyanakkor kétségtelenül ráférne valami tudatos, szervezett nyelvművelés és magyarítás.

Szaknyelvi polémiák

A magyarság a XIX. század eleje óta megőrizte nyelvújítási hajlamát. Az idegen szavak nehezen nyernek polgárjogot, és időről-időre fellobbannak kisebb nyelvi mozgalmak. Közismert a labdarúgás nyelvének magyarítása, melynek eredményeként a mai fiatalok előtt már teljesen ismeretlenek az egykor elterjedt „szakkifejezések", mint a balhalf, a henc, az ofszájd. Ezt a nyelvújítást a foci népszerűsége tette látványossá, egyúttal hozzájárulva más sportágak nyelvének magyarosodásához is. Jó példa erre a Hartyányi Zsolt által szerkesztett kis képes szótár a kosárlabdázásról, amelyben a magyar szavak mellett tulajdonképpen nincs is idegen szó (legfeljebb idegen eredetű: passzol stb.), míg ugyanebben a sportágban a németek sok kifejezést nem is „németesítettek", náluk rebound és fast break maradt az, ami nálunk lepattanó labda, eladott labda.

A magyar számítástechnikai szaknyelv alakulását az eredeti nyelvhez való tapadás és a honosító hajlam valamiféle küzdelme jellemzi. Nem annyira szakmai tudathasadásról, mint inkább szakmai csoportok küzdelméről van szó. 1985-86-ban erről a témáról kiterjedt sajtóvita folyt, melynek során Olach Zoltán például így írt: „Számoljunk le azzal az illúzióval, ... hogy magyar nyelvünk valaha is alkalmas lesz a világméretű haladás naprakész követésére." A felhozott sok ellenérv közül az egyik legfrappánsabb egy Bessenyei-idézet[1]: „Minden nemzet a maga nyelvén lett tudóssá, de idegenen sohasem." Ekkor ez sajátos politikai jelentést takart. Az angolosságnak volt bizonyos ellenzéki attitűdje, mert a hivatalos szemlélet inkább a magyaros nyelvhasználatot támogatta. Ugyanakkor a nyelvvédelem nemzetvédő jelleget is öltött. Ugyancsak az 1986-os vitában jelent meg: „Vannak, akik egyenesen nyelvünk gazdagodásának minősítik a minél több internacionalista kifejezés meghonosítását. Ez az álláspont a legveszélyesebb, mert a magyarságtudat elhalványodásához, a nemzeti nihilizmus terjesztéséhez vezet".[2] Ma az ilyesmi közhelyszerű politikai megnyilvánulás, de akkoriban valószínűleg nem jelent volna meg másmilyen szövegkörnyezetben.

A sajtópolémiák tükrözték a nyelvújítási igényt, de nem kapott hangot az a terminológiaápoló munka, amely a szakmát szinte kezdettől fogva végigkísérte. A terminológia megalkotásának és állandó karbantartásának kézenfekvő eszközei a szótárak. (A továbbiakban szótárnak nevezünk minden lexikográfiai művet, tehát lexikont, enciklopédiát stb.) Magyarországon a számítástechnika a 60-as évek második felében vált önálló szakmává, és elég hamar megjelent az első számítástechnikai szótár is, kifejezetten a terminológia megalkotásának céljából. Az IFIP nemzetközi információfeldolgozási szervezet (International Federation of Information Processing) készített egy 2940 szakmai kifejezést felölelő értelmező szótárt, hogy a szervezet tagjai az információcsere hatékonysága érdekében egységes értelmezéshez igazodjanak. Ennek 1968-ban kiadott magyar fordítása tekinthető az első hazai számítástechnikai értelmező szótárnak.[3] A fordító az előszóban megfogalmazta a magyar számítástechnikai terminológia kialakításának tendenciaként azóta is érvényes alapelveit:

— A lehetőséghez mérten törekedjünk a szakkifejezések magyarítására.

— Idegen szavak átvételekor inkább a latinos formát válasszuk.

— Olyan angol szavak esetében, amelyek magyarítására vagy latinos átvételére nincs mód, a fonetikus átírást alkalmazzuk.

Elvi és strukturális tekintetben e mű folytatásának tekinthető a Számok kiadásában megjelent két szótár. Az egyik a fogalmak köré csoportosított értelmező szótár[4], a másik önálló gyűjtésű, 5 nyelvű fordítói szótár[5]. Ezek már kevésbé magyarosak, a honosítás sajátos megoldásaként általánosan alkalmazzák az angol szavak fonetikus átírását. A következő nagy jelentőségű mű a Műszaki Könyvkiadó Számítástechnikai kislexikona[6] 1973-ban jelent meg. Mintegy 1600 fogalmat dolgozott fel egy német nyelvű forrásmű alapján, amely — miután szintén követő nyelven készült — struktúrájában és értelmezéseiben letisztultabb, mint az eredeti angol nyelvű szótárak. A német minták hozzásegítették a fordítókat, hogy kiegyensúlyozottan magyaros terminológiát alkalmazzanak. Másik jelentős alkotás 1973-ban az Akadémiai Kiadónál megjelent Műszaki értelmező szótár sorozat Számítástechnika című

kötete[7]. Erősen elméleti és matematikai (számítástudományi) jellegű. Terminológiai szempontból sajátos benne, hogy rengeteg személynévből alkotott főnevet tartalmaz (elsősorban matematikai eljárások neveit).

Szabványok és szokások

A Magyar Szabványügyi Hivatalnál 1976-ban kezdődött el a magyar számítástechnikai terminológia szabványosítása. Az eredmény két formában jelent meg: *Az adatfeldolgozás alapfogalmai* című szabványban (MSZ 7788) és *Az adatfeldolgozás többnyelvű szótára* szabványban (MI 7798). Maguk a szabványok is honosítással jöttek létre, a minta a nemzetközi szabványosítási szervezet (ISO) 2382 számú sorozata volt. A terminológia szabványosítása elvileg nagy hatású, az akadémiai szabályoknál erősebb nyelvszabályozó módszer. Az előzőleg esetlegesnek tekinthető szóhasználat „törvényerőre" emelése azonban magában rejti a megcsontosodás lehetőségét is, bár ennek veszélyét csökkenti, hogy a szaknyelv használói a szabványt nem veszik igazán komolyan. Pedig a szabványt elég széles körű szakmai vitával készítették elő, amelynek során a „nyelvőrök" közmegegyezésre törekedtek[8].

Az MSZ 7788 messzemenő figyelembevételével készült 1985-86-ban a *Mi micsoda magyarul a számítástechnikában?* című kis értelmező szótár[9], amelyre már hatott az a körülmény, hogy a mikroszámítógépek megjelenésével a számítástechnika közhasználatúvá vált, és különösen az ifjúság körében terjedt el. A *Mi micsoda...* a mikroszámítógépes tematikára koncentrált, és határozott nyelvművelő koncepciója volt. Utalásrendszerével próbált mindenkit a magyaros terminológia használatára szoktatni (például a több változatban előforduló nevek közül az értelmezés annál szerepel, amelyet a szerzők helyesnek tartottak). A címszavak angol megfelelőjének feltüntetésével és egy angol-magyar szószedettel kétnyelvű szótárként is használható.

1986-ban a Novotrade jelentetett meg egy kis méretű angol-magyar szótárt (összeállította Homonnay Péter). A tenyérben elférő kötet 2400 fogalmat tartalmaz. Kétnyelvű szótárként egyes esetekben a magyar megfelelőt közli a címszó mellett, máshol — ennek híján — körülírja az angol szó jelentését. Sok köznyelvi szó is szerepel benne, ugyanis a szerző célja az volt, hogy a szakszövegek olvasásához adjon segítséget. A 90-es évek nagy szótára a Novotrade kiadásában megjelent ún. *Oxford-szótár.* Ez ismét fordítás. Sajátossága, hogy a meghatározásokat az angol címszavakhoz rendeli, így inkább a fordítók munkáját és a szakszövegek olvasóit segítő mű.

A 90-es évektől a könyvkiadás piaci jellegének erősödésével már sok helyi szótárkezdeményezés bontakozott ki. A legfrissebbek közül — az önreklám szándéka nélkül — megemlítem a magam gondozta MicrosoftPress *Számítógép-szótárt,* mely fordítás 1999-ben jelent meg, és nemcsak frissességével, hanem méretével is kitűnik, több mint 7600 szó értelmezését és angol-magyar megfeleltetését tartalmazza.

A szótárak törzsanyagának összehasonlításából megállapíthatjuk, hogy az angolos és a magyarító tábor küzdelme nem volt elég módszeres és következetes. Például egyes kifejezések — hogy, hogy nem — kívül maradtak a hitvitákon. Ilyen például maga a *szó* (mint tárolási egység), melyet mindenki és mindig magyarul használt, vagy a *rekord,* melyet hasztalan próbáltak bárminek (mondatnak stb.) lefordítani.

Az informatika és az anyanyelv viszonyában új helyzet alakult ki, amikor szélesebb felhasználói rétegek megnyerése érdekében a nagy nemzetközi cégek nem kevés energiát és pénzt fektettek termékeik nemzeti nyelvű változatainak elterjesztésébe. (Sajnos többet, mint amennyit maguknak a programoknak a honosításába.) A lokalizálási törekvések gyártók és forgalmazók szerint tagolódtak, és ez szaknyelvi „dialektusok" kialakulásához vezetett. A koordinálatlanság hátrányosan érintette a magyar nyelvi normák érvényesítését és a nyelvművelő munkát is. Éppen ezért a továbbiakban beszámolok egy olyan nyelvfejlesztő kezdeményezésről, amellyel talán változtatni lehetne a jelenlegi széttagoltságon.

Az egység lehetősége

A Magyar Tudományos Akadémia kezdeményezte[10] egy angol-magyar informatikai szótár összeállítását, amely alapja lehet a magyar informatikai terminológia egységesítésének. Az 1998 októberétől 1999 márciusáig terjedő rövid határidő és a rendelkezésre álló anyagi fedezet csak annyit tett lehetővé, hogy rögzítsük a helyzetet, és azt sem a teljesség igényével. Az elkészült szótár magában foglalja a fellelhető terminológiai törekvések jelentős részét. Lényegében változtatás nélkül belekerült a Microsoft és az IBM lokalizációs anyaga, a MorphoLogic és a Szak Kiadó terminológiai gyűjteménye, valamint az elmúlt 30 év több más jelentős terminológiai munkája. Az eredmény a feltárt terminológiai anyag (alig felülbírált) gyűjteménye, a források megjelölésével. Terminológiai döntéseket nem kívántunk hozni, mivel sem általánosságban, sem a konkrét esetekben nem látható át az eredmények elfogadtatásának és elterjesztésének lehetősége, mechanizmusa.

Az anyag felhasználásának módja még nem dőlt el, de véleményünk szerint az egységes szakmai terminológia elterjesztésének leghatékonyabb módja az lenne, ha közvetlenül a szaknyelv aktív használóit vennénk célba. Az MTA-nak ezért arra kellene törekednie, hogy meggyőzze az informatika nyelvi „gazdáit", a könyvkiadókat, lapkiadókat, a szoftverek honosítását végző műhelyeket, hogy törekedjenek az egyeztetett szóhasználatra és az egységes terminológiai elvek alkalmazására. A közreműködés lehetőségét szervezetten kellene biztosítani, a gyakorlatban például úgy, hogy az összegyűjtött anyagból létrejönne egy folyamatosan karbantartott, szabad felhasználású, az interneten keresztül is elérhető adatbázis, amelybe a korábbi kifejezések módosulásai és a teljesen új fogalmakra adott nyelvi ajánlások meghatározott filológiai kritériumok (forráskritika, nyelvhelyesség, terminológiai minőség) alapján kerülnének bele. Az érintettek egyetértése révén ezt az adatbázist a szerzők, a szerkesztők, a dokumentátorok, a fejlesztők előbb-utóbb normaként használnák.

Kis Ádám
szak@matavnet.hu

Jegyzetek:

[1] Magyarság, 1778, idézi Csiffáry Tamás A visszahúzó szaknyelv c. cikkében. Magyar Nemzet, 1986. február 15. 14. old.

[2] Antalffy Gyula: A számítógép nyelve. Magyar Nemzet, 1986. január 11. 14. old.

[3] Sándor Géza szerk.: Az IFIP-ICC információfeldolgozási értelmező szótára. OMF, Bp. 1968.

[4] Ruisz-Síklaky szerk: A vállalatirányítási számítógép-alkalmazás fogalmainak értelmező szótára. Számok, 1972.

[5] Ruisz Rezső, Síklaky István: A vállalati számítógép-alkalmazás több nyelvű szótára. Magyar-német-angol-orosz-eszperantó. Számok, 1971.

[6] Tomcsányi Gyula szerk.: Számítástechnikai kislexikon. Műszaki Könyvkiadó, Bp., 1973.

[7] Frey-Szelezsán: Számítástechnika. Műszaki értelmező szótár. Akadémiai Kiadó, Bp. 1973.

[8] Sima Dezső: Számítástechnikát — magyarul? Magyar Nemzet 1986. február.

[9] Kis Ádám szerk.: Mi micsoda magyarul a számítástechnikában? Tömegkommunikációs Kutatóközpont, 1986.

[10] A megbízást az MTA Kutatásszervezési Intézete adta a Nemzeti Stratégiai Kutatási Program keretében, A magyar nyelv jelene kutatási témájaként.

Nevet adni egy szimbólumnak

Hogy hívják a(z) @ jelet?

Nemcsak számítástechnikai szavak és kifejezések megfelelőit kell más nyelvekben megtalálni, hanem egyes importált jelek elnevezését is. Ilyen például az angol nyelvterületen minden kereskedő és könyvelő számára magától értetődő @ egységárjel, az at. Annak ellenére, hogy a latin „ad" elöljáróból származik, csak napjainkban terjedt el világszerte, és nem is eredeti angolszász funkciójában, hanem mint az elektronikus levelek címzésének jellegzetes „figurája", a címzett nevét a gépazonosítótól elválasztó határolójel.

Nem könnyű a(z) @ írásjel sorsa, amikor olyan környezetbe érkezik, ahol az angol nyelvterületen betöltött funkciójáról tudomást sem akarnak venni. Toldalékoló nyelvekbe az „at" prepozíciót kifejező sűrítmény eleve nehezen vihető át, nálunk sem honosodott meg. Pedig latin örökségünk részeként évszázadokon át ismerték és használták. Ráadásul angol kiejtése egybeesik a szintén latin eredetű, de azelőtt sokkal szélesebb körben használt & (és) kiejtésével. Mai elnevezésének legfőbb támpontja ezért inkább különleges külalakja, az mozgatta meg a névadók fantáziáját.

Nem tudom, hogy a magyarban miért éppen a *kukac* lett a legnépszerűbb neve. Legfeljebb sejtem, mert amikor megpróbáltam másokkal leíratni, még a számítástechnikusok többsége sem tudta azt betűszerűen megrajzolni, pedig naponta az orruk előtt van több változatban is. Csak valami gubancra, összetekeredett hernyóra vagy kukacra emlékeztek, és nem is tudatosult bennük, hogy ez egy szabályos, kézírásos „a" betű, amelynek „outputja" nem a következő betű „inputja", hanem a betűt majdnem teljesen körbezáró vonal. Leírásához mindössze annyit kell megtanulni, hogy az „a" betű végződésénél a körvonalat balra és felfelé kell elindítani, és persze az egésznek a méretét hozzáigazítani a többi betűéhez.

Azokban az országokban, ahol a névadók figyelmét jobban megragadta a @ jel szerkezete, gyakori a *majomfarok* elnevezés. De emellett még sokféle becenevet találtak ki, rendszerint egy országon belül is többet. Kíváncsiságból már-már elkezdtem böngészni az egyes nyelvek számítástechnikai szószedeteit, amikor eszembe jutott egyetemi oktatóim bölcs intelme, hogy mielőtt bárminek a kutatásába belefognánk, nézzük meg a szakirodalmat, mert majdnem biztos, hogy előttünk azt a témát másvalaki már feldolgozta. Bingo! Feleki Zoli karikaturistánk lelt rá elsőként Scott Herron írására a http://www.herodios.com honlapon, amely ismerteti a kukac névhonosítási verzióit az egyes nyelvekben. Érdekes ez a kis összeállítás, amely mellesleg a @ jel történetéről szóló tanulmány első részének van feltüntetve, tehát remélhetőleg folytatása is lesz...

Nyelvenként a(z) @

Angol és amerikai angol. Hivatalos elnevezése a *kereskedelmi jel* (commercial-a, commercial-at, mercantile symbol, commercial symbol), hétköznapibb neve az *at* és az *at sign* (egységárjel), de emellett tucatnyi egyéb becenév ragadt rá magában az angolban is. Ezek közül a *darabja* (each) és a *körüli* (about) még szintén az egységár irányából közelít, de a többi már más nyelvekhez hasonlóan a külső jegyeket ragadja meg. A legtöbb az *örvény* valamelyik megnyilvánulását fejezi ki (cyclone, vortex, whorl, whirlpool), de van *majom* (ape) *csiga* (snail), *macska* (cat), *tekercs* (scroll), *rózsa* (rose) és *káposzta* (cabbage) is.

A továbbiakban csak azokra a nyelvekre térek ki, amelyekben a hivatalos kereskedelmi at mellett vagy helyett más megnevezést is használnak (és amelyeket Scott Herron összegyűjtött).

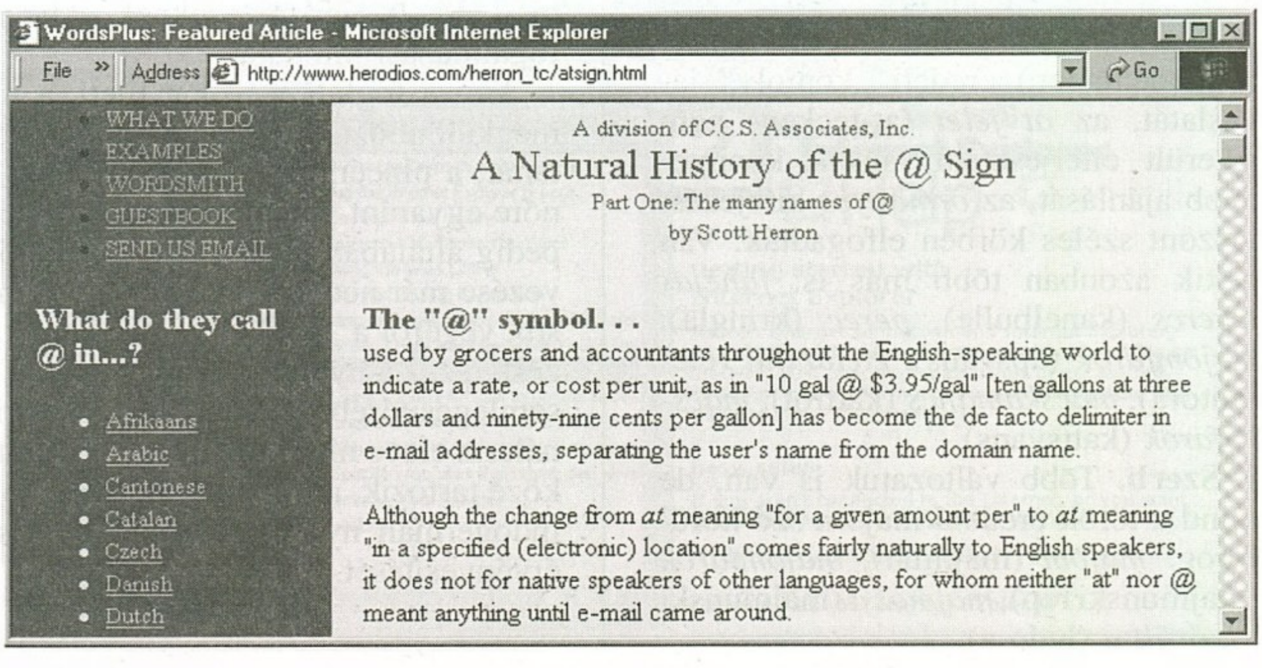
Afrikaans. A dél-afrikai holland telepesek leszármazottainak nyelvén a @ neve *majomfarok* (aapstert).

Arab. Az e-mailt használók körében gyakori az „at" arab megfelelője (fi), de felbukkant a *fül* is (othon) a fülkagylóra emlékeztető rajzolat alapján.

Cseh és szlovák. Egyaránt az összetekeredésre figyeltek fel, és azt a *heringkarikákkal* fejezték ki (zavinac).

Dán. Két elnevezés gyakori, az egyik az *alfa-jel* (alfa-tegn), a másik az *elefántormány* (snabel), de előfordul a *malacfarok* is (grisehale).

Finn. Északi rokonaink hangot is asszociáltak a jelhez, ezért a *macskafarok* (kissanhanta) elnevezés mellett macskanyávogásnak is hívják: miau, miumau, miuku. (Nyávogójelek: miau merkki.) Van azonban náluk *majomfarok* (apinanhanta) és *egérfarok* (hiirenhanta) elnevezés is.



Francia. Leggyakoribb francia elnevezése arobase, amely feltehetően a spanyol arroba rokona, és kifejezetten csak a @ jelet jelenti. Van emellett persze más neve is: *tekert a* (a enroule), *csiga* (escargot) vagy *kis csiga* (petit escargot).

Héber. Ebben a nyelvben is a tekeredő alak az elnevezés forrása: az egyik a *csiga* (shablul), a másik a *rétes* (shtrudl).

Holland. Az eredeti holland megnevezés korrekt leírás, de kissé nehézkes: *egy a egy hurokkal* (een a met een slinger), nem csoda, hogy hamarosan felváltotta a *majomfarok* (apestaart), a *majomfarkinca* (apestaartje) és a *hurkolt majom* (slingeraap), de van emellett *göndör a* (a-krol, a-krul) és *kis majomhere* (apeklootja) is.

Japán. A japánok az „at" jelet vették át, de az saját hangszerelésükben úgy szól, hogy attu maaku.

Koreai. A koreai nyelvbe a @ mint *csiga* vonult be (dalphaengi).

Kínai. Angol „at" jelként történő emlegetése ragadt meg a kantoni (hongkongi) nyelvben, viszont a tajvaniaknál inkább az *egérke* (xiao lao-shu) és az *egér jel* (lao shu-hao) használatos.

Lengyel. Legelterjedtebb a *majom* (malpa), a *cica* (kotek) és a *disznófül* (ucho swini).

Német. A németben a leggyakoribb a *majomfarok* (Affenschwanz) és a *pókmajom* (Klammeraffe), de szokták *fülként* (Ohr) is emlegetni.

Olasz. Az olaszok a *csigával* társították a jelet (chiocciola).

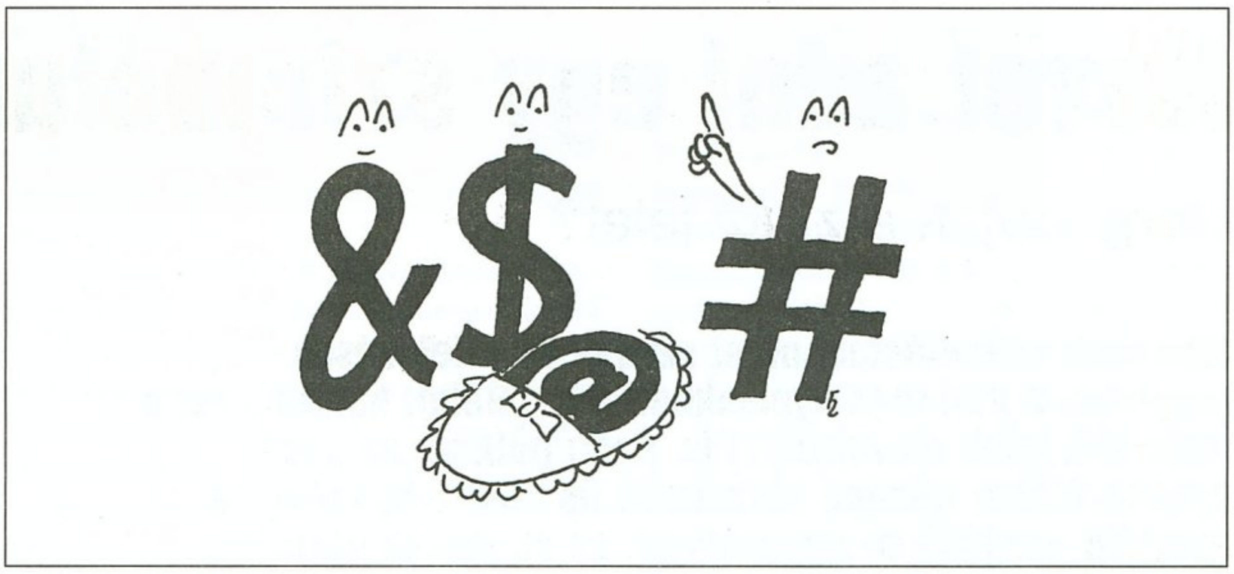
Norvég. Az angol „at" mellett helyi változat a *malacfarok* (grisehale) és a *göndör alfa* (krøllalfa).

Orosz. Hivatalos neve a *kereskedelmi a*, de elterjedt a *kiskutya* (szobácska), továbbá előfordul a *majom* (obezjána) és *pacsni* (pljúska) elnevezés is.

Portugál és spanyol. Mindkettőben egy régi súlyegység neve, az arab eredetű „arrova" ragadt rá.

Svéd. A svéd nyelvi tanács (Svenska Spreknemnden) eredeti „komoly" javaslatát, az *at jelet* (at-tecken) nem sikerült elterjeszteni, másik könnyedebb ajánlását, az *ormányos* kifejezést viszont széles körben elfogadták. Van nekik azonban több más is: *fahéjas tekercs* (kanelbulle), *perec* (kringla), *majomfarok* (apsvans), *elefántfül* (elefantora), *macskamancs* (kattfot), *macskafarok* (kattsvans).

Szerb. Több változatuk is van, de mind a török eredetű majom szó körül forog: *majom* (majmun), *majomfarok* (majmunski rep), *majmos a* (majmunski a), *őrült a* (ludo a).



Szlovén. Egyesek szerint az afna elnevezés a német *majomfarok* (Affenschwanz) praktikusan lerövidített változata, bár nem lehet teljesen véletlen, hogy mivé rövidítették, ugyanis az afna eredeti jelentése agyoncicomázott, túlöltözött nő...

Thai. Végre egy velünk rokonlelkű elnevezés: *tekergő kukacszerű karakter* (ai tua yiukyiu).

Török. A *fül* (kulak) honosodott meg, német kapcsolataik révén annak Ohr formájában is. Javasolták ugyan az at (ejtsd et) átvételét, de az törökül lovat jelent.

Ha a fenti elnevezések közül most kellene nevet választani, valószínűleg ismét a kukacra szavaznék. Teljesen szubjektív dolog ugyan, de második helyen nekem a *perec* a legszimpatikusabb, harmadik helyen pedig a *csiga*.

A fenti összeállítás azért is érdekes, mert egyetlen karakteren keresztül bepillanthatunk egy kicsit a honosítás színes világába. A legfőbb tanulság pedig az, hogy amikor az angol szakkifejezésekre megfelelő magyar változatot keresünk, érdemes szétnézni más nyelvek háza táján is.

Faklen Pál

Nemek, nevek és névmások

A magyarítás során gyakran panaszkodunk az átültetés nehézségeire, és többnyire olyan következtetésre jutunk, hogy „nekik" bezzeg könnyebb. Scott Herron honlapján találhatunk egy másik érdekes tanulmányt (The Wordsmith), mely arra hívja fel a figyelmet, hogy az előnyök és a hátrányok „ott" is karonfogva járnak.

A fogalmazást tömörebbé teszi, ha egy nyelvben a névmások, mutatószók és egyes főnevek önmagukban kifejezik a férfi vagy női nemet, de amikor éppen általános érvényű megfogalmazásra van szükség, emiatt komoly nehézségeket okoz nekik a nemi megkülönböztetés elkerülése. Ez az álláshirdetésekben a legkényesebb, ám a probléma sokkal kiterjedtebb, gondoljunk csak a műszaki leírások, használati utasítások tömegére, amelyekben lépten-nyomon felbukkan a probléma, hiszen egy készülék használóját sok esetben ösztönösen is egyes szám harmadik személyben nevezik meg. Scott Herron ezeknek a helyzeteknek a megoldására ad ötleteket, fogalmazási mintákat.

Ami a foglalkozásokat illeti, a magyar nyelvben is van nemek szerinti megkülönböztetés, de ez sokkal könnyebben áthidalható, mert például a tanár, a pincér, az ügyvéd stb. elég általános ahhoz, hogy idővel férfira és nőre egyaránt vonatkozzon, a női nemet hangsúlyosan kifejező -nő toldalék pedig általában elhagyható. Az elnőiesedett szakmák egy részének megnevezése már nem ennyire versenysemleges, és az óvónő, a védőnő, a varrónő stb. végéről a „nőt" elhagyva nem kapunk a férfiakra is érvényes megnevezést, de ez a nyelvi jelenség éppen a férfiakra nézve kirekesztő. A névmások semlegesítésébe fektetett munkát viszont megtakaríthatjuk, mert azoknak nálunk eleve nincs férfi vagy női „attribútuma". Ez tehát azon ritkább esetek közé tartozik, amikor mi vagyunk jobb helyzetben, és az angol (meg a többi indogermán nyelv) készteti egy kis agytornára az egyenjogúság (formai) érvényesítését.

Mellőzött megoldás az Ablakban

Kiaknázatlan lehetőség a DOS/Windows honosulására

Jogos és természetes igény, hogy anyanyelvünkön lehessen kezelni a programokat. Hogy egy alkalmazás mennyire tehető jól kommunikáló eszközzé, az nem független a futtatókörnyezettől és annak honosításától. A következőkben azt szeretném áttekinteni, hogy a munkahelyeken leggyakoribb operációs rendszereken, a DOS –> Win9x platformokon milyen lehetőségekkel élhetünk. A programok és megoldások csak illusztrálás céljára kiragadott példák.

Az alkalmazások egyre inkább azok munkáját hivatottak megkönnyíteni, akiknek a számítástechnikai ismeretei igencsak gyérek. Nem rendszergazdák, nem programozók, és általában nem is a számítástechnika szakangoljának ismerői. A DOS esetében a futtatókörnyezet valójában meglehetősen egyszerű, mivel az alkalmazások futtatását a command.com koordinálja. Ennek a fájlnak a lokalizálása jóval megelőzte a nemzeti verziók hivatalos megjelenését.

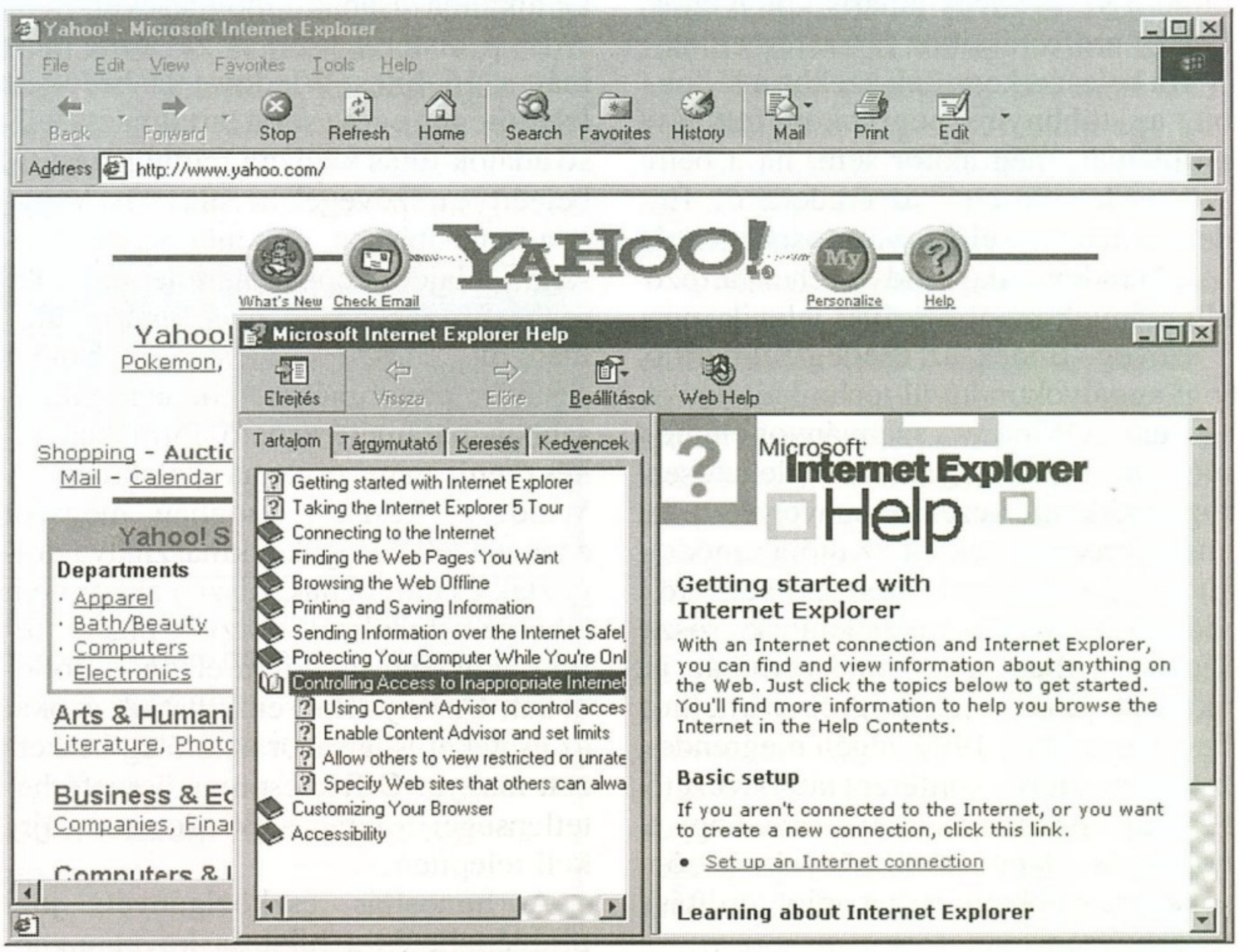
Az egyszerű lokalizálás eszközigénye, hála a közvetlen lemezírás, olvasás lehetőségének, nem volt nagy. Elegendő volt egy, a valós bájtértékeket megmutató szerkesztő, amilyen a HackerViewer vagy akármelyik diskeditor, például a PC-Tools lemezmanipuláló editora. Ha a DOS-ra írt alkalmazásokat szerettük volna honosítani, más esetekben sem volt jobb megoldás. De az egyetlen igazán jó szövegátírási eszköz a program készítőjének kezében volt. Módosítania kellett a forráskódban a szövegelemeket, majd újrafordítani a teljes programot. Azonban a karakteres felületet alkalmazó programok képernyőképét jelentősen befolyásolta a DOS-ban definiált karakterkészlet.

Hiába készítette el valaki a hazánkban elterjedt ékezetesített kódkiosztások (CWI, 850, 852) valamelyikén a szöveget, ha a futtatás másik kiosztású környezetben történt. Ilyenkor furcsa karakterek jelentek meg a magyar szövegekben meglehetősen gyakori ékezetes betűk helyén. Ismét elő kellett szedni a már említett editorokat, és a bináris állományban saját gépünknek megfelelően újraírni a szövegelemeket. Feltéve, hogy megvoltak az eszközök egy tekintélyes méretű fájl átnézésére, hogy a programfájl nem volt tömörítve és egyirányúsítva — például PKLite-tal vagy más hasonló programmal. És még a CRC-t figyelő vírusirtókra is oda kellett figyelni.

Voltak továbbá programok, amelyeket DOS-ra készítettek ugyan, de saját grafikus felülettel láttak el. Ezekben átrajzolhatjuk a karaktereket... Elvileg. Mert a gyakorlatban a grafikailag kevésbé igényes megoldások esetén a legegyszerűbb továbbra is a szöveg közvetlen kiíratása marad, például a Turbo C++ és Turbo Pascal fejlesztőkészletek BGI-rendszerével. A gyárilag adott betűkészletekben azonban ékezetes betűink nincsenek ott teljes létszámban, így a bináris fájlban megtalált szövegek bináris átszerkesztése sem jelent megoldást, és inkább a programozó fantáziáján múlik, hogy milyen trükkel tudja például a hosszú „Ű" betűt megjeleníteni.

A megoldásra maga a Borland kínált eszközt. A BGI Toolkit segítségével magam is elkészítettem a Turbo Pascalban használható alapvető fontkészletek magyarított változatait. (Azok annak idején az Új Alaplap floppymellékletén meg is jelentek.) Visszatekintve ezekre, a probléma csak az volt, hogy az ékezetes betűk meglehetősen „egzotikus" kódokra kerültek, amennyiben a szabványos karakterkiosztásokat vettük alapul. De működtek, megoldva legalább a program által kiírt üzenetek függetlenségét a gépen éppen használt karaktertáblától.

Ugyancsak a megjelenítés, a pixelszintű tetszőleges kirajzolás előnyeit van hivatva kihasználni az MS-DOS legelterjedtebb grafikus héja, a Windows is. A Microsoft a 16 bites Windows verziókban vezette be a TrueType betűkészleteket a rugalmasan méretezhető kiírások céljára, a karakterkezelésben pedig az „egyszerűség" jegyében saját karakterkiosztásokat definiáltak. Ennek előnyeiről és árnyoldalairól az évek során sokan sokféle érvet felsorakoztattak. Nem kívánunk most ez ügy-



ben állást foglalni, tekintsük inkább át a Windows programokban alkalmazott leggyakoribb szövegkezelési eljárásokat. A Windows programok készítésekor a programozónak továbbra is adott a lehetősége, hogy magába a forráskódba „drótozza bele” a kiírandó szövegeket. Ekkor azonban minden egyes lokalizálási vagy hibajavítási lépésben magát a forráskódot kell editálni, megkeresve a szövegek összes előfordulási helyét, és valamennyit egyenként kijavítva vagy átírva.

A Windows programokban azonban adott a lehetőség, hogy külön fájlokban tároljuk a szövegeket, és azokat a fordító kapcsolja hozzá a futtatható kódhoz. A megoldás előnye a memóriagazdálkodás javításán kívül az, hogy a stringtábla szövegsoraira csak egy-egy azonosító hivatkozik a program forráskódjában. Javítás esetén tehát elegendő egy példányban, ezekben az ún. erőforrásfájlokban elvégezni a módosításokat. A felhasználói alkalmazások ilyetén átírása nem is okoz problémát, ha birtokában vagyunk a forráskódnak és a megfelelő fejlesztőeszköznek. Ezek azonban legtöbbször csak saját fejlesztés esetén vannak meg, és az ilyen programok a megfelelő kódtábla birtokában magyarul beszélnek ugyan, de a Windows saját vezérlőelemei, például gombjai, dialógusablakai, és az alaprendszerhez tartozó menüelemek továbbra is őrzik az eredeti nyelvi beállításokat. A fejlesztőeszközökkel kapott erőforrás-szerkesztőkkel viszont lehetőségünk van belenézni a bináris állományokba, így mind a futtatható fájlok, mind a szabványos bináris könyvtárak (DLL) erőforrásaihoz hozzáférhetünk.

Ha beleszerkesztünk ezekbe a fájlokba, az többnyire nem okoz futtatási problémát, még akkor sem, ha a beírt szöveg hossza eltér az eredetiétől. Érdemes azonban elővigyázatosnak lenni, ha a Windows alaprendszeréhez tartozó állományok meghívásával jelenik meg a szöveg. Ennek az esetleg felmerülő jogi aggályokon kívül technikai okai is vannak. A Windows szabványos elemeinek jó része ugyanis meglehetősen rugalmatlanul kezeli a szövegeket, és ennek eredményeként az utólag módosított, az eredetinél hosszabb vagy rövidebb szövegek megjelenítése rosszabb esetben még rendszerhibát is okozhat. Erről a jelenségről számoltak be a hazánkban 1999 végén megrendezett honosítási konferenciát bevezető sajtótájékoztatón is, utalva arra, hogy a honosítás ilyen körülmények között sokszor inkább művészet, mint fordítási tevékenység.

A Windows helyi verzióinak elkészítése szintén a programozók feladata. Esetünkben a Microsoft munkatársaié. A módosított szövegeket tartalmazó erőforrásfájlokat és kódrészleteket nekik kell a forráskód birtokában beépíteni a helyi verziók kódjába. Eredményként egy belső verziószámában és legtöbbször fizikai méretében is eltérő állományt kapunk. Természetesen azokban az esetekben is, amikor az alkalmazói program kerül honosításra. A futtatható állományok, például a WINWORD.EXE számos más, eredetileg a Windows által biztosított szolgáltatást is meghív, és ennek eredményeként — példánknál maradva — az MS Word telepítőkészlete tartalmazza a módosított rendszerállományokat is.

Amennyiben ezeket eltérő nyelvi verziók összeépítésekor (angol Windows + magyar Word vagy viszont) engedjük felülírni, akkor könnyen a Windows elhalálozásához vezető eredményre juthatunk. Vagy a működőképesség határát súroló instabil rendszerhez. De az eredeti állományok megtartásakor sem biztos a hibátlan működés, mert az utólag telepített programok a kisebb verziószámmal rendelkező komponenseket gyakran nem hajlandók megfelelően kezelni. Ez vezetett arra az elterjedt és jól megalapozott álláspontra, hogy különböző nyelvi verziókat tartalmazó rendszereket ne telepítsünk egymásra. Annál is inkább, mert a Microsoft is csak a nem otthoni használatra szánt Windows 2000-ben ígérte a DLL-pokol (DLL-hell) végleges felszámolását a komponenstechnika alkalmazásának kibővítésével.

Ez a história azért is zavaros, mert már a 16 bites Windows is beépített lehetőségek egész sorát tartalmazta külső adatok futás közbeni feldolgozására, beleértve a szövegek kezelését is. Vagyis a honosításhoz, a menüpontok átírásához tulajdonképpen nem lenne szükség a forráskódra, sem a bináris állományok átszerkesztésére, hanem „csak” a programokat kellene rugalmasabban paraméterezhető formában elkészíteni, vagy olyanná átalakítani. A Windows készítői azonban mégsem ezeket a lehetőségeket kihasználva dolgoztak és dolgoznak, hiszen akkor nem lehetne a lokalizált verziók piacra kerülésekor újra meg újra eladni a dobozokat. Bizonyítékul említhetjük azokat az eseteket is, amikor az utólag beszerzett magyar Office csomag összeférhetetlenségei folytán a Windowst is újra kell telepíteni.

A honosítás eszközigényét, jogi problémáit és a hibázás következmé-



nyeit tekintve legkeményebbnek a DOS esetében ismertetett eljárások tekinthetők. Ezt követik azok a lehetőségek, amelyek a kód újrafordításával teszik csak megoldhatóvá a szöveges elemek módosítását.

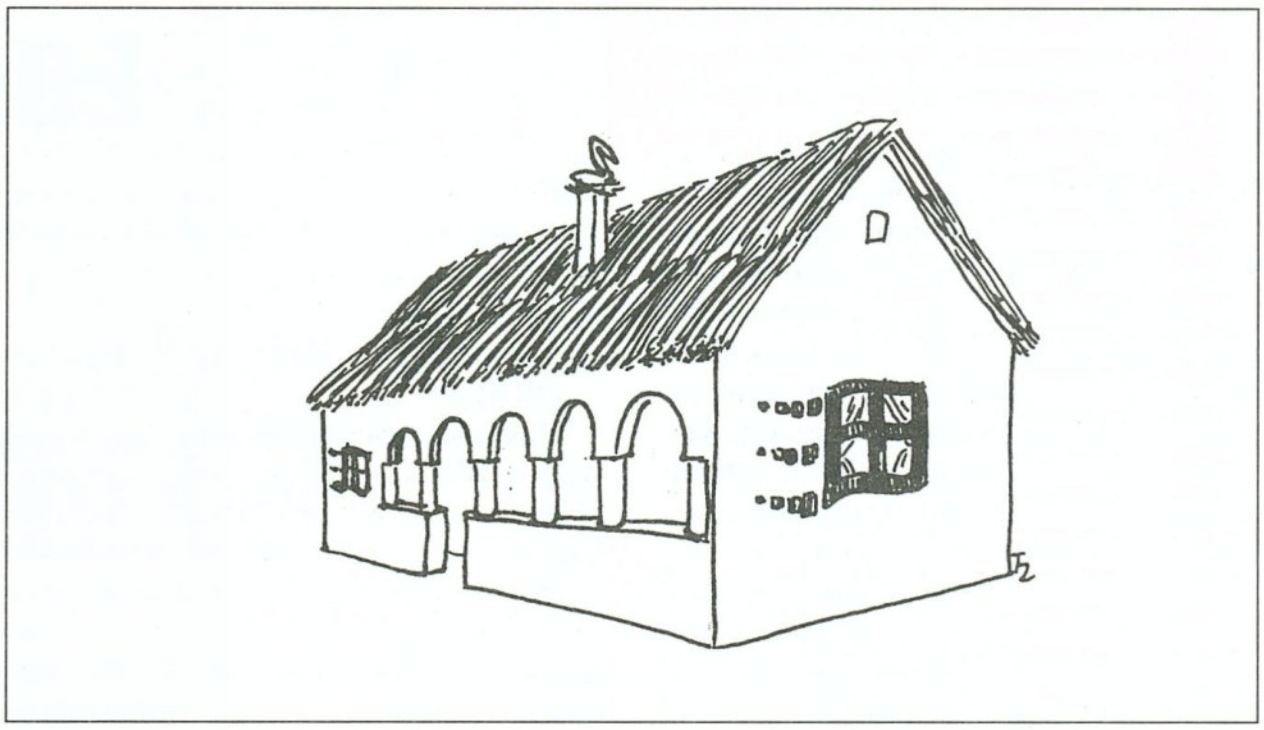
A szabad szövegkezelés megvalósítása ellen felhozott érvek többsége nagyon gyenge lábon áll. Azok az érvek, amelyek a szabványosságot és a minőségi követelményeket hozzák fel, nem igazán megalapozottak. Ennek kapcsán említettük már a különböző alkalmazásokban így is, úgy is megjelenő heterogenitást, de ezen csak minden érdekelt által elfogadott szakszótárak segíthetnének.

A szerzői jogok védelmét hangsúlyozó érvek sem valami meggyőzőek. A rosszul sikerült honosítások házi barkácsolással végzett javítása sokkal inkább sértheti a szerzői jogot, mint az, ha a nyelvi környezet bárki által szabadon átformálható, és a futtatórendszer számára erőforrásként értelmezhető elemek formájában lenne beépítve a programokba. A programozói energiákat pedig fektessék inkább az alkalmazások futtatásának optimalizálásába. Ezzel elkerülhetők lennének a szerzői jogokkal védett gyári elemek ütközéséből eredő károk is, amelyekért ezen elemek szerzőinek érdekes módon nem akaródzik felelősséget vállalni. Hogy így a megjelenő ablakos elemekbe bárki bármit

beleírhatna, az való igaz. De nem is kellene a szabad paraméterezhetőséget mindenre kiterjeszteni. Például a hibaüzeneteket tényleg rögzítve kell tartani, hogy a hibajelenségek emlegetésekor végre mindenki ugyanarról beszéljen. Különösen, ha azok a honosítás révén a hiba okáról vagy jellegéről semmilyen részletes információval nem szolgálnak, az egyszerű ember abból sem tud meg többet, mintha az angol eredetit olvasná, a rendszergazda, a programozó, a tesztelő számára pedig kötelező a szakmai angol legalább ilyen szintű ismerete.

A windowsos elemekkel kapcsolatban szokták felhozni, hogy a beillesztendő szöveghez nincs mindig elegendő hely a befogadó ablakban. Ez azonban csak akkor igaz, ha az ablak mérete kötött. Tapasztalatom szerint azonban Delphiben, CBuilderben és Visual Basicben sem okoz problémát olyan dialógusablakok létrehozása, amelyek a megjelenítendő szöveg függvényében képesek átméretezni magukat. A lefordított kódok az önszabályozó elemekkel együtt valóban hosszabbak lennének, de a ma használatos 32 bites Windowsok és alkalmazásaik terjedelmét ez lényegesen nem növelné. Ez nem…

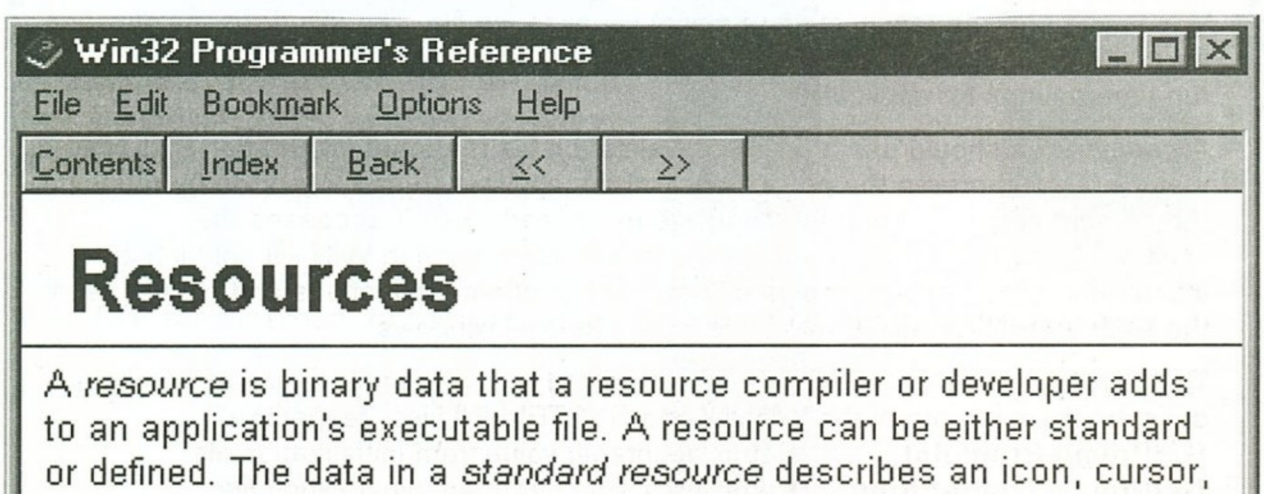
Ennyi kacskaringó után nézzük meg közelebbről, miként lehetne megoldani a többnyelvű szövegek tárolását, a szövegbevitel felszabadítását a Windows alaprendszerének újraalkotása nélkül. A merevebb megoldásoknál felhasználhatjuk az erőforrásokat is erre a célra. A fizikai kód növelése árán semmi akadálya, hogy egyetlen bináris fájlban tároljuk a különböző nyelvi verziók szövegeit. A nyelvfüggő hivatkozásokat utána már csak az erőforrások szövegelemeinek szelektív behívására kell felhasználni. Nem jelentene túl nagy programozási pluszmunkát, de hordozná a méretnövekedésen kívül mindazokat a hátrányokat, amelyek a bináris fájlok verziókövetésénél jelentkeznek. Nem túl nagy alkalmazásoknál, és nyelvérzékeny esetekben mégis jól jöhet, hogy egyetlen menüutasításra a teljes nyelvi környezet lecserélhető, ahogy a Morphologic által készített 16 bites szótárprogramban ez már megfigyelhető volt.



Sokkal jobb lenne azonban, ha mindenki maga tudná szövegfordításaival ellátni a programokat. A Windows erre is több megoldást kínál. Az egyik a Registry felhasználása szövegtárolásra. A jelenlegi programok telepítésével is rövid idő alatt több Mbájtosra hízó központi adattár kezelésére a Windows API (Application Programming Interface) beépített eljárásokat tartalmaz. Részletes leírásuk megtalálható a fejlesztőeszközök mellé járó win32.hlp állományban. Segítségükkel egyszerű függvényhívásokkal kérdezhetjük le a Registryt, az adott kulcshoz tartozó információt. Ezek képezhetnék a Windows szabványos elemein megjelenő feliratokat, amennyiben a Windows készítői ezzel a megoldással élnének. Szintén beépített függvények gondoskodnak új kulcsok készítéséről és az adatok elhelyezéséről a központban, és ez a fejlesztőnek gyakorlatilag tetszőleges mennyiségű adat ottani raktározására adna lehetőséget. A módszer előnye az lenne, hogy kombinálva az önméretező elemekkel, többé nem okozna gondot, hogy az egyik DLL magyar, a másik angol, német stb. nyelvű elemeket jelenít meg. Akár kevert nyelvű dialógusablakok is lehetnének.

Ugyancsak előnye lehetne ennek a megoldásnak, hogy a lokalizáláshoz készülhetne olyan segédprogram, amely levenné a Registry manuális szerkesztésének (regedit.exe) terhét és kockázatát a felhasználóról. Sorba lehetne menni a szabványos, akár angol nyelvű meghatározásokon, és bekérni a lokalizált kifejezéseket. Ez ugyan munkaigényes, de ha a program kimenete nem közvetlenül a Registry, hanem egy külön fájl, akkor elegendő azt egyszer legenerálni. Az említett potenciális fájlkimenet is ki van találva.

A Microsoft már a 16 bites világban rögzítette a központi adattár számára emészthető *.REG fájlok szerkezetét. Amennyiben programunk ilyennek a kialakítására vetemedne, a gombok és más szabványos elemek szövegének elkészítését egyszerre lehetne megoldani — akár országosan is. A honosítás a bináris állományok újrafordítása és a forráskód átírása nélkül megtörténhetne. Ebben az esetben a hibák javítása is lényegesen egyszerűsödne, mivel a szöveges *.REG fájlban egyszerű szövegszerkesztővel lehetne a módosításokat elvégezni. Vagy akár képzeletbeli segédprogramunkat lehetne keresési és javítási funkciókkal ellátni, miközben a központi javítások nem verziófüggő



DLL-készletekként utaznának, hanem egyetlen floppyn elférő szövegben.

A Registryben tárolt szövegek módszere ellen szól, hogy amennyiben az említett segédprogram nem készül el, a manuális szerkesztés nem veszélytelen. Ugyanakkor mélységesen csalódnék a shareware és freeware programok készítőinek hadában, ha nem készítenék el azt a Microsofttól függetlenül is! Szintén előnytelen, ha a szöveges elemek felpuffasztják a központi adattár állományait, mert több Mbájtnyi méretük jelentősen lelassíthatná a Windows egyébként sem villámgyors indulását, és esetenként a működését is. Az sem jó, hogy egy-egy újratelepítéskor az alapoktól célszerű építkezni — kidobva a túlhízott system.dat + user.dat párost —, és a honosítást elölről kell kezdeni, de ezen a szöveges verzió tárolása segíthet.

A Registryben való tárolás hátrányainak kiküszöbölésére vissza kell menni egy kicsit a múltba. Már a Windows 1.x is tartalmazott .INI kiterjesztésű fájlokat a különböző adatok, elsősorban a Windows működési paramétereinek tárolására. Ezen fájloknak belső szerkezete azóta is szabványos, és a Windows 3.x-ig magának az alaprendszernek is a legfontosabb tárolómechanizmusát jelentette. Az egyes alkalmazások szintén ezt használták, aminek eredményeként a Windows könyvtárban felszaporodtak a szöveges állományok. A belső szerkezetében folyamatosan változó Registry szerepének előtérbe kerülését részben éppen ez indokolja. Ez mégsem jelenti azt, hogy a win.ini és a system.ini szereptelenné vált volna, túl a továbbra is .INI-vel dolgozó alkalmazásokon. Ezeknek a fájloknak a kezelésére már a Windows 3.x is beépített függvénykészlettel rendelkezett, s használatukat a modern fejlesztőeszközök (Delphi) szintén beépített eljárásokkal fedik le.

A fejlesztőeszközökben is megnyilvánuló, a Windows API-n alapuló beépített támogatásnak köszönhetően az .INI fájlokban tárolt nyelvi adatokkal mindazok az előnyök megvannak, amelyeket Registrynél említettünk. A honosítás továbbra is központilag, könnyen szerkeszthető szövegállománnyal volna megoldható, amelyet mindenki saját munkakörnyezete szerint módosíthatna, szükség szerint javíthatna. Ugyanakkor nem terhelnénk a központi adatbázist az összességében terjedelmessé hízó szövegelemekkel. A nyelvek közti kapcsolás is egyszerű lenne, mivel a programnak csak azt kellene megmondani, hogy melyik nyelvhez melyik .INI tartozik. Ezekben pedig szabványosított fejezetcímekkel lehetne gondoskodni arról, hogy például melyik gombtípushoz melyik szöveg (OK, YES, DA, IGEN stb.) tartozik az adott nyelven. Referenciaként akár az eredeti szöveget is tartalmazná, mint az alábbi (elképzelt) példában: ...

[OKButton]

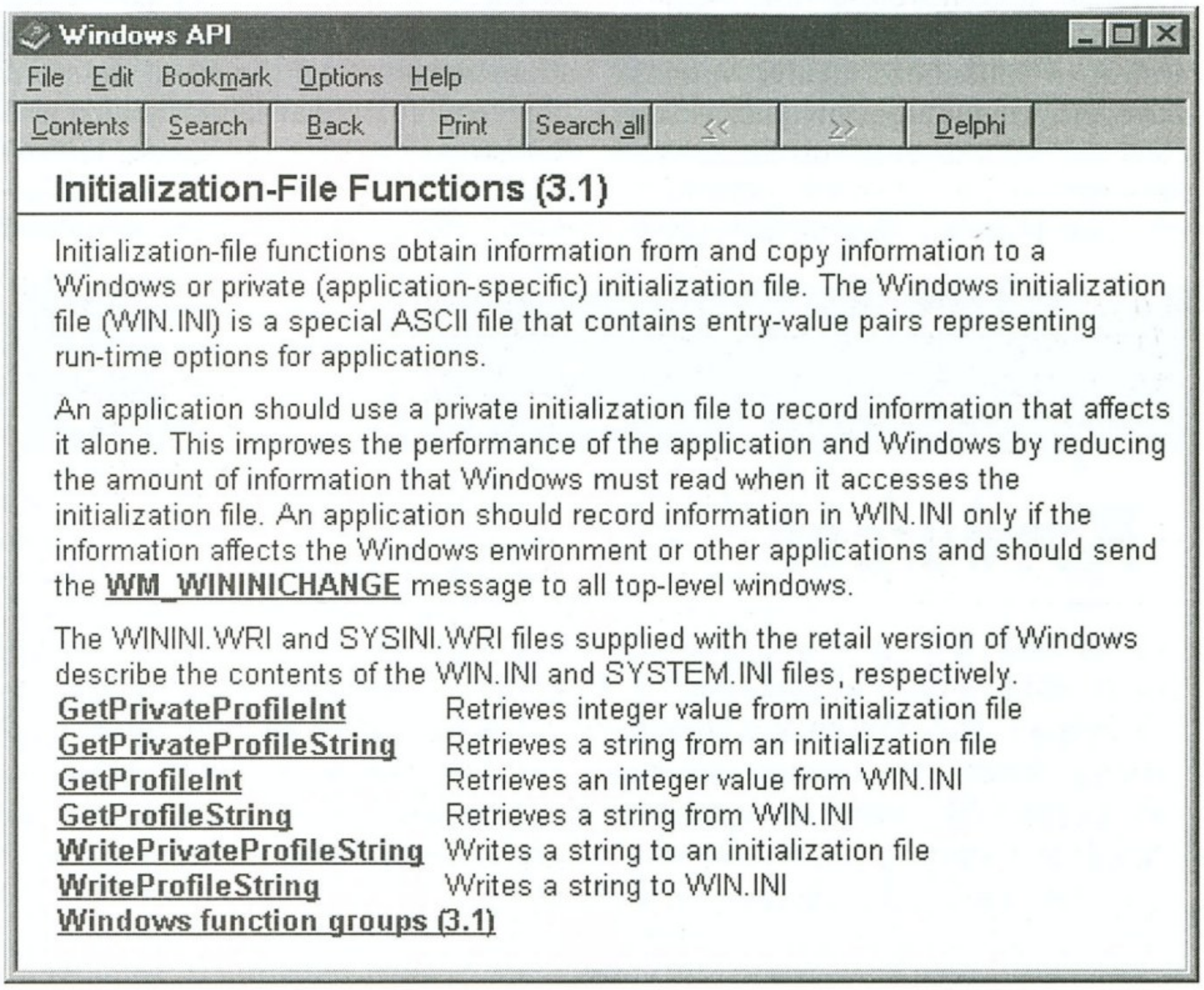
OK=Igen ...

Hasonló mechanizmus figyelhető meg a Windows Commander menükezelésében. Ahogy azt januári számunkban is említettük, ennél a windowsos Norton Commander klónnál a különböző nyelvű menüfájlokat külön szövegállományok tartalmazzák. Szerkesztésükkel szabadon testre szabhatjuk a különböző menüelemek elhelyezkedését éppen úgy, mint a szövegüket. A menüfájlban alkalmazott belső struktúra emlékeztet a Borland erőforrás-szerkesztőivel létrehozható menüleíró fájlokéhoz. Valószínűleg nem véletlenül, mivel a Windows Commander fejlesztése, ahogy azt szerzője még a 16 bites verzió megjelenésekor publikálta, nagyrészt Delphivel történik. Ez is szabványos szerkezet, de az egyes alkalmazások készítésekor (elvben) bármilyen megoldást kitalálhatunk.

Az alaprendszer megvalósításaira azonban célszerűbb lenne egy szabványos és publikus megoldást találni. Szabványosat, hogy egységes programozási, honosítási lehetőségek nyíljanak meg a külső fejlesztők előtt anélkül, hogy ezzel instabillá tennénk a Windows futását. Publikusat pedig, hogy bárki bármikor módosíthassa a gyárilag esetleg nem túl jól sikerült honosításokat, és a javításhoz, módosításhoz ne kelljen binárisan matatni a fájlokban akkor is, amikor az nem létkérdés. A Windows alapalkatrészeinek készítésekor ehhez fel kellene használni azokat a lehetőségeket, amelyeket maga a Windows API kínál. Mert bár az MS Office 2000 és a Windows 2000 esetében az eddigi tapasztalatok szerint kiküszöbölték a DLL-ek összeférhetetlenségét, és a nyelvi csomagok telepítése is egyszerűbb, a szabad hozzáférésről eddig nem érkeztek hírek.

Azt azonban sajnos látnunk kell, hogy a fent jelzett finomhangolási lehetőségek esetén a Microsoftnak már nem lenne extraprofitja a honosított verziókból. Ennek hasznát a felhasználók élvezhetnék a világ minden országában, ami ellentétes a Microsoft filozófiájával. Ahogy jeleztük, egyetlen floppyval megoldhatnánk a teljes honosítást a megfelelő .REG, .INF vagy .INI fájl felhasználásával. Nem lehetne újra és újra eladni a teljes, néha több CD-s telepítőkészletet, amelyekben a szövegmódosítások hatására, vagy a honosításra használt verziók különböző volta miatt a hibák száma sem feltétlenül csökken. Legfeljebb reménykedhetünk abban, hogy egyszer mégiscsak a Windows alapkoncepciójának megfelelően, a Windowsban meglévő adottságok alapján válik természetes „helyi" műveletté a honosítás a nem angol anyanyelvű országokban.

Simay Endre István



Honosítás BeOS alatt

Karakterkódoktól a grafikus felületig

A honosítás felületesen szemlélve nagyon egyszerű: csak le kell fordítani a programban található szövegrészeket, és kész. Elég azonban emlékeztetni a Windows magyar változatára, hogy e téren milyen könnyen elkövethetők egészen nevetséges hibák is.

A honosítási problémák fő forrása a sok nyelv sokféle karaktere. Az angol nyelv betűkészletéhez alkalmazkodó 7 bites, 128 karakteres kódtáblában sok nyelv speciális karakterei nem férnek el, de a 8 bites, 256 karakteresbe is csak a „protekciósak" juthattak be. Ezen először úgy próbáltak segíteni, hogy kódlapok (code page) egész sorát definiálták a nyelvekhez és nyelvcsoportokhoz, ami végül sokkal több káros mellékhatással járt, mint eredetileg gondolták (kódlaptöltögetés, konverziós problémák, kompatibilitás hiánya, áttekinthetetlen fontkészlethalmaz stb.). A következő lépés a 16 bites, 65 535 karaktert tartalmazó Unicode volt. Ez minden karaktert 2 bájton tárol, ami a latin betűs nyelvek esetében elég nagy pazarlás, ráadásul nem lehet kompatibilis a használatban lévő sokféle ASCII kódkiosztással.

A kompromisszumot az UTF szabvány hozta, amelynek kódjai a Unicode részhalmazát képezik. A BeOS is az UTF-8 alapján kezeli a karaktereket. Mindaddig, amíg a karakterek értelmezhetők ASCII kódtáblával (0x7f érték alatt), a program azt használja. Ha jön egy karakter, amely ebbe már nem fér bele, akkor a program ezt a legfelső biten jelzi, és saját kódolásával ábrázolja két vagy több bájton. Itt most nem térek ki a részletekre, a lényeg az, hogy az UTF-8 felülről kompatibilis az ASCII-val, tud ábrázolni karaktereket egytől tizenhat bájt hosszig, így kódolható vele az összes nyelv karakterkészlete. A BeOS rendszerben minden sztring UTF-8 kódolású. Ez azt jelenti, hogy az ablakok neve akár japánul is lehet, és a Standard C Library rutinokig minden fel van készítve az UTF-8 kódolású sztringekre.

Infrastruktúra

Ahhoz, hogy jobban átlássuk a BeOS adta lehetőségeket, érdemes megismerkedni annak alapvető infrastruktúrájával. A BeOS rendszerben például kulcsszerepet játszik a szálak közötti kommunikáció. Pár szóban ismertetem ezzel összefüggésben a legfontosabb fogalmakat.

Team: Ez egy program közös címtartománnyal, más néven process. Több szál (thread) alkot egy teamet.

BLooper: Mivel a BeOS API-ja C++-ban készült, adódik az ötlet, hogy legyen egy C++ osztály, amely külön szálon fut, és végtelen ciklusban üzeneteket tud fogadni és feldolgozni. Amikor létrehozzuk, ez az objektum elindít egy új szálat, és ezen az új szálon fogadja a neki címzett üzeneteket.

BHandler: Az effektív feldolgozást végző osztály. Ha egy BLoopernek küldünk egy üzenetet, megadhatjuk, hogy a looperhez kapcsolt handlerek közül azt melyiknek szánjuk, és a rendszer meghívja a címzett handler virtuális MessageReceived( ) metódusát. Egy BLooper maga is BHandler, azaz képes üzeneteket feldolgozni egymagában is, hozzácsatolt handlerek nélkül.

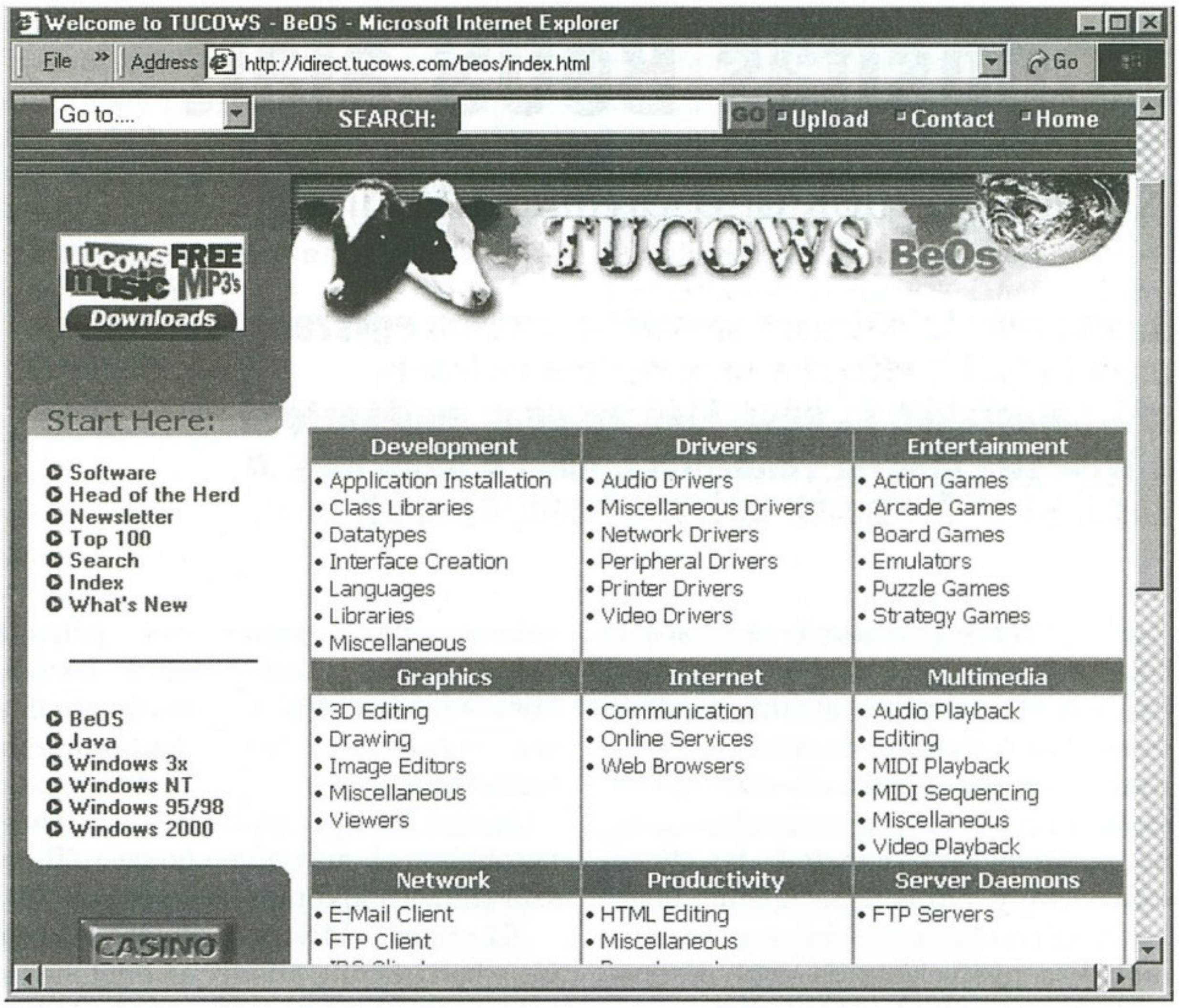
BMessage: Az üzeneteket reprezentálja. Ez összetett feladat, gondoljunk csak arra, hogy egy üzenethez kapcsolódó adatok milyen sokfélék lehetnek (például egérgomb üzenet esetén a gomb kódja egész szám, a koordinátája BPoint objektum...). De maradjunk a leírásnál: egy BMessage objektum tartalmazhat névvel ellátott bármilyen típusú adatot, amelyet később a név alapján kikereshetünk. Ezenfelül egy BMessage objektum „kilapítható" (flattenable), ami azt jelenti, hogy le tudja menteni a benne lévő adatokat egy folyamba, és azokat később vissza is tudja olvasni onnan. Tehát alkalmas eszköz arra, hogy üzenetet küldjünk egy másik szálnak vagy programnak (akár egy másik gépre!), vagy ami miatt nekünk most fontos, hogy lementsünk egy BMessage-et egy fájlba.

A fentiek alapján tehát tudunk üzeneteket küldeni és fogadni, és az üzenetekbe bármilyen adatot „becsomagolni". Innen már továbbléphetünk a grafikus felület felé.



Egy BLooper tökéletesen megfelelő arra, hogy egy ablak alapjául szolgáljon: tud üzeneteket fogadni, külön szálon fut, és nincs belőle jelen egyszerre annyi példány, hogy ésszerűtlen lenne mindegyiknek külön szálon futnia. Ezért a BWindow a BLooperből lett származtatva. Ennek sok hasznos következménye közül az egyik legfontosabb, hogy BeOS-ban nem találkozik az ember beragadt ablakokkal, amelyeket arrébb sem tud rakni.

Az ablakba való rajzolást segíti a BHandlerből származtatott BView osztály, amely ezáltal képes a rendszerüzeneteket és a mások által küldött üzeneteket is fogadni, ha hozzá van csatolva egy BWindow-hoz. (Ez olyan, mintha több különböző méretű papírlapot egymásra rakva az egyes lapok csak a belőlük kilátszó felületért lennének felelősek, és ezek együttese adná meg az ablak teljes képét.) A BView-kba rajzolhatunk, és emellett azok még üzeneteket is fogadhatnak. Az egész BeOS Interface Kit erre az architektúrára épül. Minden gomb, tekerő és egyéb a BView-ból lett származtatva, és minden üzenetüket (például egy billentyű lenyomását) alapértelmezésben annak az ablaknak küldik, amelyikhez csatolva lettek, de a címzett természetesen át is állítható.



A GUI

Nézzük meg, hogyan épül fel egy BeOS-os program grafikus interfésze. Ha létre akarunk hozni egy ablakot, akkor örökölni kell a BWindow osztályt, és meg kell írni a MessageReceived( ) metódust, hogy reagálhassunk az ablaknak küldött üzenetekre.

Ezután ablakunkra aggathatunk mindenféle BView-t, amelyek bizonyos esetekben üzeneteket fognak küldeni (például egy BButton, ha megnyomták), és kaphatnak is üzeneteket (átméretezés, háttérbe kerülés, egérmozgatás, gombnyomás...). De csinálhatunk teljesen autonóm view-t is, ha a view-hierarchiában az alatta elhelyezkedő összes view-nak az üzeneteit átirányítjuk egyenesen neki (ezek ugye alapértelmezés szerint az ablaknak mennének, ami a fa gyökere). Ezáltal a view teljesen független lesz attól, hogy melyik ablakban helyezkedik el, sőt még attól is, hogy melyik programban van! (De ne rohanjunk előre, erről később még ejtünk pár szót.)

A GUI-t illető másik fontos probléma a sokkarakteres nyelvek (például a japán) beviteli módja. Japánul nem lehet a klasszikus módon gépelni, mert ahhoz igen nagy klaviatúra kellene, ezért egy karakter általában több billentyűleütésből áll össze. Ehhez speciális, rendszerszintű támogatás szükséges. Erre szolgál a BeOS-ban egy „Input Method". Ez egy kis nyílt API-jú segédállomány (hasonló mint egy DLL), amelyet az Input Server tölt be, és az egész rendszerben mindenhol használható. Amikor egy szöveges mezőbe begépelünk valamit japánul, megnyílik egy kis ablak, és ez segíti a bevitelt. Nekem nem állt módomban kipróbálni, mert nem tudok japánul, de akikkel beszéltem, azok mind dicsérték.

Archiválás

Nézzük meg jobban a Bmessage nyújtotta lehetőségeket. Mivel mindenfélét belegyömöszölhetünk, tökéletesen alkalmas rá, hogy adatokat tároljunk benne, például objektumokat archiváljunk. Ez azt jelenti, hogy az objektum minden tulajdonságát elmentjük, aminek alapján később újra létrehozhatunk egy teljesen ugyanolyan objektumot. Minden BLooper, BHandler (és ezáltal minden belőle származtatott osztály) a BArchivable osztályból származik, és tartalmaz egy Archive( ) metódust, amelynek az a feladata, hogy az objektum pillanatnyi állapotát lementse egy BMessage-be. Ha írunk egy osztályt (amely extra attribútumokat is tartalmaz azokon felül, amelyeket ősétől örökölt), akkor meg kell írni azt a néhány soros Archive( ) metódust, amely saját attribútumainkat is lementi egy BMessage-be (például egy BButton esetén a BView attribútumain kívül a gomb neve stb.), és kell írni egy egyszerű konstruktort, amely az objektum létrehozásakor visszaolvassa egy archív BMessage-ből ezeket az attribútumokat. Egy BMessage tartalmazhat BMessage-eket is, hasonlóan, ahogy a view-k tartalmazhatják egymást. Ez lehetőséget nyújt „deep" (mély) archiválásra, amikor egész ablakot archiválunk az összes általa megjelenített BView-val együtt egyetlen BMessage-ben.

Röviden: le tudjuk menteni grafikus objektumaink pillanatnyi állapotát egy BMessage-be, majd azokat onnan újra létrehozhatjuk. Ez programunkban két elegáns sorra egyszerűsödik. Lementés: win –> Archive(message, deep = true). Létrehozás: win = reinterpret_cast<BMessage *>(instantiate_object (message)).

Mire van szükség még ahhoz, hogy „megrajzolhassuk" programunk grafikus felületét? Egy másik programra, amellyel felépíthetünk egy ablakot, vagy bármely más grafikus objektumot, majd archiválhatjuk azt egy BMessage-be. Ezt az archívot a mi programunkba beolvasva átadjuk ablakunk konstruktorának, és az létrehozza az egész ablakot úgy, ahogy a tervezőprogrammal megrajzoltuk. Egy ilyen program az Interface Elements, amit a BeOS-os tábor egyik elismert magyar programozója, Mezei Attila írt (http://w3.datanet.hu/~amezei). Egy kicsit tovább játszva a gondolattal, újra le is menthetjük ezt a létrehozott interfészt, amikor kilép a

programunk, ezáltal ablakunk ott fog megnyílni, és akkora lesz a következő indításnál, mint a kilépés pillanatában volt.

Mindez úgy tűnhet, mintha ágyúval lőnénk verébre, pedig nem. A BeOS szkriptelési lehetőségeivel ugyanis át lehet színezni például egy futó program nyomógombját a programon kívülről, anélkül, hogy a program írója valaha is gondolt volna ennek lehetőségére. És ha a grafikus felület kilépéskor is le van mentve, akkor a program következő indításakor ez a nyomógomb megtartja új színét, amelyet mi állítottunk be kívülről. Megtehetjük ezt például egy shell szkriptből — a szintén Mezei Attila által írt „hey” nevű programmal. Nagyon leegyszerűsítve csak annyit, hogy ez BMessage-eket küld az általunk választott program bármelyik objektumának, és arra az reagálhat. A BeOS szkripting nem tévesztendő össze a klasszikus shell vagy perl szkriptekkel, amelyekből elérhető a BeOS szkriptelési szolgáltatása. Hangsúlyozom: mindez anélkül működik, hogy a program írója bármire is figyelt volna az archiválás egyszerű feladatán kívül.

Honosítás

Hogyan kapcsolódik ez az egész a honosítás témájához? Nagyon közvetlenül: amikor le akarunk fordítani egy programot valamilyen nyelvre, csupán be kell tölteni az editorba a grafikus interfész archívját tartalmazó BMessage-eket, átírni az ablakok, nyomógombok és egyebek neveit, majd lementeni az új nyelvet tartalmazó archívot a többi közé. A program készít nekünk egy listát az archívokról, és felkínálja a lehetőségeket, majd az általunk választottból építi fel az egész grafikus felületet. Így nem okoz gondot, ha valamilyen nyelven tizenöt betűből áll az a szó, amely angolul csak hárombetűs. Ha egy szöveg nem fér el ugyanakkora helyen, a honosítás során könnyedén átrendezhetjük a grafikus felületet.

De az sem okoz problémát, hogy a Cancel magyarra legalább háromféleképpen fordítható, attól függően, hogy milyen összefüggésben használjuk. A nyomógombok sorrendje is megváltoztatható a honosítás során, és ami a legfontosabb: a program kódjához hozzá sem kell nyúlni, bárki bármelyik programhoz (!) csinálhat saját fordítást a program forrása nélkül. Ez persze átgondolt API-t igényel, amellyel nem fordul elő, hogy a program magyarul kérdez valamit, majd válaszlehetőségként felajánlja, hogy Yes és No.

Ha már itt vagyunk, elgondolkodott már valaki azon, hogy a Windows ezt miért csinálja? Én akkor jöttem rá, amikor az egyetemen megtanították, hogyan kell DialogBoxot nyitni... Láttam, hogy az API ezt szinte a fülébe súgja a programozónak. Amigás múltammal sehogy sem értettem, miért kell végigolvasnom minden requestert, hogy megtudjam: azt kérdezi-e a program, kilépjen-e, vagy azt, elmentse-e a változásokat. Mert ugye itt az igen és a nem épp az ellenkezőjét váltja ki a programból. De akkor miért nem teszi fel minden program legalább ugyanazt a kérdést?

A BeOS fent említett eljárása persze csak a GUI elemeknél működik ilyen egyszerűen, ha a program más módon is kommunikál (például naplót generál), és a többnyelvűség igénye arra is vonatkozik, akkor saját szótárat kell fenntartania. Különböző segédprogramokkal ez azonban megoldható, a többi operációs rendszerben megszokotthoz hasonlóan.

És ami még mögötte van

Ha már így megismertük a BeOS által nyújtott lehetőségeket, nézzük meg, mire képes még ez az architektúra. Először egy kis kitérő: minden BeOS rendszeren futó programnak van egy egyedi azonosítója (signature). Ez egy MIME string, mint például „application/x-vnd.Be-StyledEdit”. Ennek alapján a program a másodperc tört része alatt megtalálható a merevlemezen, ugyanis a BeOS fájlrendszere lehetőséget ad az indexelésre bizonyos attribútumok alapján. Ez látványosan felgyorsítja a keresést.

Most nézzük meg, mire van szükség ahhoz, hogy egy archivált objektumot újra létrehozzunk. Kell hozzá először is az objektum adatait tartalmazó BMessage. Azután kell az objektum kódja, ami persze mindig rendelkezésre áll, ha egy teamen belül dolgozunk. De egyszerűen megoldható az is, hogy az archivált objektum létrehozható legyen egy másik teamben is, csak hozzá kell adni az objektum adataihoz annak az exe fájlnak a signature-jét, amelyben az osztály kódja található. Ha a rendszer nem találja az archivált objektum osztályának kódját, akkor megkeresi a megadott signature alapján, hogy melyik exéében vagy más végrehajtható fájlban található, és betölti azt, mint egy DLL-t.

E módszer lehetőségeit kihasználva megtehetjük, hogy egy ablakhoz kívülről hozzáadunk egy (akár szövegszerkesztő bonyolultságú) view-t, amelyről az ablak csak annyi tud, hogy egy BView-ból örökölt osztály, közben pedig az egész egy szövegszerkesztő. Ez úgy képzelhető el, hogy megragadjuk például az óraprogram kijelzőjét (ami egy BView-n vagy egy abból származtatott osztály), és átdobjuk egy másik ablakba. A BeOS észreveszi, hogy az üzenet archivált objektumot tartalmaz, kikeresi az óraprogram exéjét, betölti annak a programnak a címtartományába, ahová dobtuk, és létrehozza a kijelzőt, amely szépen mutatja majd az időt, miközben az eredeti óraprogramból akár ki is léphetünk. Ezeket a másolatokat replikánksoknak hívják.

Ha a program a fent leírt archiválást használja, akkor a következő indításnál már ott lesz benne az óra kijelzője, anélkül, hogy a program írója valami erőfeszítést tett volna ennek érdekében. Egy kis odafigyeléssel csinálhatunk akár olyan szövegszerkesztőt is, amelynek dokumentumai replikánsokat tartalmazhatnak. Elküldve barátunknak egy replikánst tartalmazó dokumentumot, az nála is teljes funkcionalitásával jelenik meg. Ez hasonló lehetőségeket nyújt, mint a windowsos OLE, csak sokkal egyszerűbben.

A távlatok

Ez a cikk csak csak az architektúra néhány vonását tárgyalta, és nem azt, hogy mit nyújtanak a kész programok. A programozók munkáiban ezek (egyelőre) még nem nagyon jelennek meg. Én is csak most kezdek átállni ilyen megoldásokra, mert először meg kellett ismernem az eszköztárat ahhoz, hogy átlássam a benne rejlő lehetőséget. És itt-ott még hiányoznak hozzá a megfelelő eszközök is.

A BeOS azonban rohamosan fejlődik, gombamód szaporodnak a cikkek a programozásával kapcsolatban. Az elkövetkező hetekben fog megjelenni (vagy a cikk megjelenésekor már meg is jelent) a BeOS R5, amely otthoni használatra ingyenes lesz. Szóval elő a böngészőt, és nyomás a free.be.com címre! A www.be.com címen megtalálható a teljes programozói dokumentáció, a hardverkompatibilitási lista, és még sok egyéb. Friss hírek szerezhetők be a BeOS háza tájáról a www.benews.com, a www.begroovy.com, a www.beoscentral.com és a www.beforever.com webhelyeken.

A BeOS Games portál címe www.begames.com, a BeOS shareware programoknak a lelőhelye pedig a www.bebits.com.

Lendvai Attila
101@inf.bme.hu

A gépi fordítás ma is kihívás

Számítógépes nyelvészeti projektek Japánban

A gépi fordítással kapcsolatos elképzelések szinte az első számítógépekkel egyidejűleg megjelentek. A kezdeti lelkesedést azonban kiábrándulás követte. A túlzott várakozás felkeltésében tagadhatatlanul nagy szerepe volt a szenzációhajhász sajtónak is. Később ugyan reális terveket is kidolgoztak, de a számítástechnika viszonylagos fejletlensége miatt a lehetőségek jóval elmaradtak az igényektől. Az elvégzett munka hozadékaként azonban kialakult egy új tudományág, a számítógépes nyelvészet, amelynek egyik távlati célja a fordítás bonyolult folyamatának gépesítése.

A számítógépes nyelvészet már számos területen bizonyította, hogy eredményei jól hasznosíthatók a gyakorlatban. Érdemes egy kicsit utánagondolnunk, mi lett az eredetileg megfogalmazott feladatokkal.

Eltérő motiváció

A gépi fordítás (GF) iránti érdeklődés az utóbbi időben Japánban felerősödött. Nehéz azonban világos képet kapni kutatásaik jelenlegi állásáról, mert az eredmények nagy része nem kerül a nyilvánosság elé. Sok energiát lekötött, hogy a korábbi fordítási kutatások zömét áttették PC-re. Közben nagy átszervezések voltak, például megerősítették és kiszélesítették azokat a projekteket, amelyeket már eredetileg is mikrogépekre, illetve kisgépekre terveztek.

Néhány évtizede az Egyesült Államok elsősorban katonai megfontolásokból foglalkozott intenzíven a gyakorlatban is hasznosítható fordítórendszerek kifejlesztésével. Japán számára ez sokkal inkább a műszaki fejlesztésnek, és a korszerű műszaki információk gyors elterjesztésének a kérdése volt, még olyan áron is, ha a sietség a minőség rovására megy.

Európában más volt a kutatások fellendülésének motivációja. Itt elsősorban a soknyelvűség és a lefordítandó anyag hatalmas tömege játszott szerepet a gépi fordítás iránti érdeklődés megélénkülésében. Az Egyesült Államokban továbbra is magas szintű kutatómunka folyik, elsősorban az egyetemeken, ez azonban elsősorban elméleti irányultságú. Európa ilyen szempontból gyakorlatiasabb, és a kutatások fő finanszírozója az Európai Unió, nyilván azért, mert joganyagait és fontosabb dokumentumait minden jelenlegi és leendő tagország nyelvére le kell fordítani.

Érdemes hangsúlyozni a lényeges különbséget Európa és Japán gyakorlati igényei között. Az idő mint tényező Európa számára sem mellékes, de sokkal kisebb a szerepe, itt a készülő fordítások értékelésében a minőségnek van prioritása. A fordítás folyamatának teljes automatizálása helyébe már csak ezért is egyre inkább a géppel segített fordítás lép, mert az interaktív rendszerek (akár hálózatokról, akár egyedi gépekről van szó) ehhez igen jó lehetőséget kínálnak. Teljesen automatizált rendszerek legfeljebb egészen szűk szakmai területen jöhetnek számításba (meteorológiai előrejelzések, többnyelvű vasúti tájékoztatás stb.).

Dokumentumértékű fordítások esetében feltétlenül ki kell küszöbölni a durva félrefordításnak még a lehetőségét is. A közvetlen gépi produktum legfeljebb nyersfordításként (előfordításként) hasznosítható, ez is inkább olyan helyeken, ahol az eredeti szöveg alapján azonnal kontrollálható a szöveg értelme. Ugyanakkor a szinte „gombnyomásra készülő" nyersfordítás is hasznos lehet például az internetezőknek vagy a műszaki információk tengerében eligazodni szándékozóknak, főleg akkor, ha utána a fordításra érdemesnek kijelölt részeket valaki pontosan le tudja fordítani.

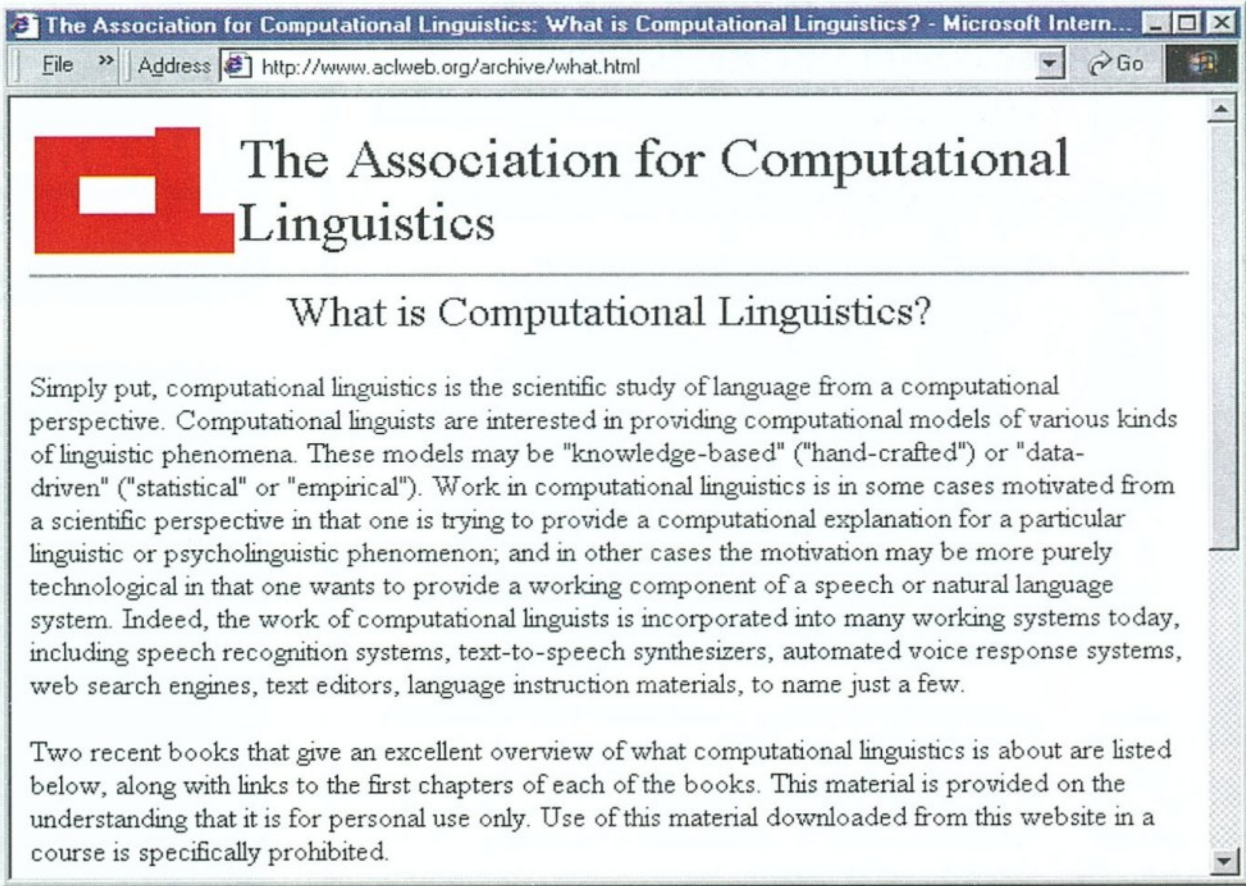
Sokan azt hiszik, hogy a japánok jól tudnak idegen nyelveket. Ez tévedés. Viszonylag kevesen és elég rosszul beszélnek még angolul is. Viszont szinte minden fontosabb szakirodalmi anyaghoz hozzájutnak saját nyelvükön. Pillanatok alatt lefordítják, a szerzői joggal nem sokat törődve, és egy-két hónapon belül ki is adják japánul. Még a szerző neve sincs mindig feltüntetve, japán átírásban sem — de ki az, aki ezt innen Európából észreveszi? Az is előfordul, hogy olyan zugkiadó neve kerül rá a kiadványra, amely kizárólag ennek a kiadásnak a lebonyolítására jött létre. A szerzői jog birtokosa tehát elég nehéz helyzetben van, ha jogait érvényesíteni akarja.

Aki mer, az nyer

Japán már nemegyszer meglepte a világot merész húzásaival, néha vakmerőnek tűnő üzleti és fejlesztési politikájával, amely sok esetben mégis eredményesnek bizonyult. Olyan dinamizmus alakult ki a 80-as évekre, amely később már szinte önmagától működő fejlesztőerővé vált.

A tágabb értelemben vett gépi fordítási kutatások legfőbb mentora Japánban az állami irányítású Csúcstechnológia Központ (Japan Key Technology Center, JKTC). Ennek kezdeményezésére indult el a „Basic Research of Automatic Translation Telephone" elnevezésű alapkutatási program, és 70%-ban költségeinek fininszírozásáról is gondoskodott. Eredetileg 7 éves időtartamra tervezték, 17 milliárd yen (kb. 110 millió dollár) költségvetéssel. A távlati cél egy teljesen automatizált szinkrontolmácsoló rendszer kifejlesztése, amely lehetővé teszi egymás nyelvét nem értő személyek (elsősorban szakemberek és üzletemberek) telefonbeszélgetését a fordítógép közbeiktatásával. A kitűzött cél tehát több mint ambiciózus.

Ebbe az állami projektbe hatalmas beruházásokkal szálltak be a telefonfejlesztésben érdekelt nagyvállalatok is. A kifejlesztendő rendszer első változata az V. Telekommunikációs Világkiállításon, Genfben, 1987 októberében mutatkozott be. Kétirányú (angol–japán, japán–angol) online gépi fordítórendszerüket működés közben mutatták be a nagyközönségnek. A Világkiállítás résztvevői élőben társaloghattak Tokióval a számítógépek billentyűzetén és


The Association for Computational Linguistics: What is Computational Linguistics? - Microsoft Intern...

File » Address http://www.aclweb.org/archive/what.html Go

The Association for Computational Linguistics

What is Computational Linguistics?

Simply put, computational linguistics is the scientific study of language from a computational perspective. Computational linguists are interested in providing computational models of various kinds of linguistic phenomena. These models may be "knowledge-based" ("hand-crafted") or "data-driven" ("statistical" or "empirical"). Work in computational linguistics is in some cases motivated from a scientific perspective in that one is trying to provide a computational explanation for a particular linguistic or psycholinguistic phenomenon; and in other cases the motivation may be more purely technological in that one wants to provide a working component of a speech or natural language system. Indeed, the work of computational linguists is incorporated into many working systems today, including speech recognition systems, text-to-speech synthesizers, automated voice response systems, web search engines, text editors, language instruction materials, to name just a few.

Two recent books that give an excellent overview of what computational linguistics is about are listed below, along with links to the first chapters of each of the books. This material is provided on the understanding that it is for personal use only. Use of this material downloaded from this website in a course is specifically prohibited.

a képernyőn keresztül. A Tokióban japánul bebillentyűzött kérdéseket Genfben angolul olvashatták, és angolul válaszolhattak rájuk. (Akkor még maga a távkonferencia sem volt olyan mindennapos dolog, mint ma.) A párbeszéd ugyan megmosolyogtatóan primitív volt, de a mögötte rejlő munka és a továbbfejlesztés lehetősége ma sem lebecsülendő. A gép képernyőjén követhetők voltak a fordítás egyes fázisai is.

A telefonbeszélgetések szinkrontolmácsolása természetesen nagyságrendekkel nehezebb feladat, de nem látszik kivitelezhetetlennek. Ne feledjük el, hogy telefonon rögtön rá lehet kérdezni arra, amit nem értünk, és az ismétlés során más kifejezéseket használva a fordítás pontosabb lehet. (Ha a beszélőket nem zavarja, az eredeti szöveg is hallhatóvá tehető.) Egy ilyen rendszernek akkor is hasznát lehet venni, ha a tolmácsolás nem tökéletes, viszont nem várható tőle hiteles fordítás, azt más eszközökkel kell kombinálni, hogy egy üzleti megállapodást előkészítsenek.

A szótárprojekt

Kevésbé látványos, de ugyanolyan fontos (vagy talán még fontosabb) célkitűzéssel indult a 90-es évek elején az a program, amely a gépi fordítás alkalmazhatóságának legkényesebb pontján akar előbbre lépni. Az elektronikus szótárak kidolgozásáról van szó, minél pontosabban definiált értelmezésekkel, jelentésváltozatokkal, és természetesen a gép számára is érthető módon formalizált felépítésben.

A projekt pénzügyi alapja hasonló, mint a szinkrontolmácsoló rendszeré. Külön kutatóintézetet hoztak létre a projekt központi szervezésére és fő feladatainak megoldására (Japan Electronic Dictionary Research Institute, EDR). Az elektronikus szótárak kutatóintézete minden bizonnyal az előirányzott 10 év elteltével sem fog megszűnni. Kezdettől fogva együttműködik számos kiadóvállalattal, és bevonták a munkába azt a fejlesztőintézetet is, amelyet eredetileg az ötödik generációs számítógépek kifejlesztésére hoztak létre (ICOT, Institute of the Fifth Generation Computer Project). Ez utóbbi projekt a 90-es évekre elhalt, a befektetett szellemi és anyagi tőke azonban nem tűnt el nyomtalanul. Az intézet szellemi és műszaki potenciálja fennmaradt, és a magasabb színvonalú gépi fordítás megteremtésére irányul.

Vargha Dénes

Magyar kód gombnyomásra

A számítógépen az írott nyelv fő támasza a billentyűzet, amelynek kezelése alapvetően a BIOS feladata. Erről könnyen meggyőződhetünk, ha kihúzzuk a csatlakozóját a gépből, mert rögtön jön a csipogás meg a hibaüzenet. Az operációs rendszer voltaképpen a billentyűzetet kezelő megszakítás segítségével olvassa ki a lenyomott gombok BIOS-kódját. Ezért lehet az operációs rendszer szintjén meghatározni a honosított billentyűkiosztást. A megszakításról lekérdezett kódot az operációs rendszer szinte tetszőleges másik kódra lefordíthatja, és gondoskodhat megfelelő megjelenítéséről. Ezért dolgozhatunk az amerikai QWERTY billentyűzeten a hazai QWERTZ (németes) kiosztással is.

A meglepetések akkor jönnek, amikor olyan programmal találkozunk, amely a billentyűzetkezelés gyorsítása érdekében megkerüli az operációs rendszert. Ilyenkor fellép például azt a jelenség, hogy a 'Yes' válaszhoz a 'Z' betűt kell lenyomni. Keletkezik is némi bonyodalom, ha valaki nem ismeri az eredeti angol-amerikai kiosztást. A régebbi programok használói nem véletlenül ragaszkodnak az amerikai klaviatúrához, inkább vállalva a magyar feliratozás kényelmetlenségeit. (A fordított jellegű feliratozás szokása nem terjedt el.)

Bosszúságot okozhatnak a nem elég körültekintően megírt programok is. A BIOS kódtáblájában 8 bites kódolásnál nincs külön hely a kurzorbillentyűknek és a funkciógomboknak. Ahogy a Shift, Ctrl, Capslock, Numlock billentyűk lenyomásakor egy bit bebillentésével éltek, ezeknél a billentyűknél is trükkhöz folyamodtak: a billentyűzetpufferbe kettős kódot visznek be. Az első egy #00, a második pedig megadja a tényleges kódot. Ez az iránybillentyűk és az F1 … F10 gombok esetében jól tanulmányozható a CD-mellékleten elhelyezett rövid Pascal forrás segítségével. Amennyiben a program nem kezeli a #00 értékű karaktert, egy kurzormozgatás könnyen ugyanazt eredményezi, mint egy betű leütése.

Univerzális megoldást erre a Windows úgy kínál, hogy elfog minden billentyűlenyomási impulzust, majd a nyelvi beállításnak megfelelően generálja a felhasználói programok számára már egységesen megjelenő virtuális kódot (VK, virtual key). A billentyűmozgáshoz tartozó VK érték a Win3.x esetében is 16 bites, abba minden billentyű belefér, amit a szintén 16 bites kódokkal operáló, de nem mindig pozitív szerepet betöltő unikód is jól demonstrál.

Simay Endre István

Honosítás

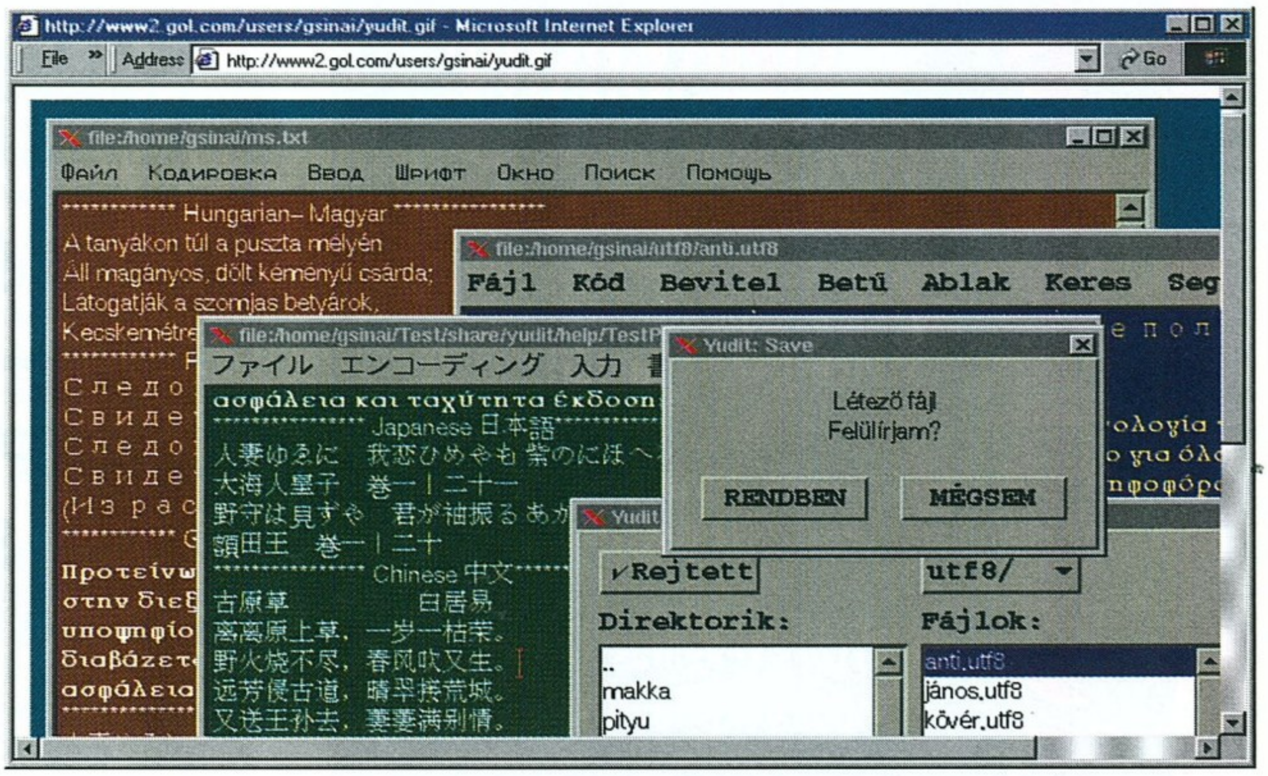
A hónap témájához több „részletkérdés” is kapcsolódik. A CD-melléklet Fókusz rovatának összeállításakor igyekeztünk a sokféle anyagból ennek megfelelően válogatni.

A honosítás egyik neuralgikus pontja a karakterkészletek kezelése. A CD-n erre vonatkozóan elhelyeztünk dokumentumokat és segédprogramokat, főképp az unikóddal kapcsolatban. (A magyarosított unikód nevet használjuk, mely az univerzális kód rövidített és köznevesített alakja. Ez talán korainak tűnik, mert ma még a Unicode inkább tulajdonnévként viselkedik. Az *Internet* kontra *internet* írásmódvitában az Új Alaplap képviselte legtovább a tulajdonnévi felfogást, de a többség már a köznévi formára szavazott, így mostani számunktól kezdve mi is átvesszük a kisbetűs írásmódot. A *Unicode* esetében legyünk mi az újítók, hisz előbb-utóbb bizonyára ennek is a köznévi és magyaros *unikód* formája honosodik meg.)

Mindenki áttanulmányozhatja azokat a részletes információkat, melyek a kiszélesített kódkészletre vonatkoznak, így érthetőbbé válik, hogy mi okoz olykor gondokat a programokban. Az unikódot kezelő alkalmazások között általában a Windows különböző verzióira készült programokat szokták emlegetni, pedig van több más is. Illusztrálásul a CD-n közreadjuk a Linuxra készült Yudit szövegszerkesztőt.

A fontkezelő programokból külön gyűjteményt bocsátunk olvasóink rendelkezésére, igaz elsősorban Windows platformra. Közöttük betűkészleteket megjelenítő, szabadon használható programok éppúgy megtalálhatók, mint a TrueType fontok szerkesztésére készült Font Creator. Ez utóbbival akár BMP formátumból is készíthetünk betűket, bár szabadon sajnos csak a program shareware változata érhető el a http://www.high-logic.com címről. Szintén a TTF betűkészletekkel kapcsolatos a Mac-es TT Converter 1.5, a PC és a Macintosh közötti konverzióhoz.

Szabadon méretezhető fontokat nemcsak a nagyok (Microsoft, Apple, Adobe stb.) készítenek. Egy gyűjtemény a http://www.freetype.org/old_index.html weboldalon is elérhető, részben forráskóddal együtt terjesztett anyagként. A FreeType 1 (így a CD-n található FreeType 1.3.1 is) egy library, mellyel mindenki maga készíthet alkalmazásokat. Igazi fontszerver csak a FreeType 2 lesz, amely jelenleg még béta állapotban van.



A honosítást elősegítő eszközök másik nagy csoportjába a fordítástámogatók, szótárak, szószedetek, kislexikonok stb. tartoznak. Igyekeztünk ezekből is érdekes és hasznos anyagokat összeválogatni. A Kvoctrain program Linuxra készült csináld magad szótáralkalmazás. A kész anyagok közül a \Techszot könyvtárban komplett és szabadon használható online szótárat nyújtunk át, míg a \Jargon könyvtár azt a világhírű gyűjteményt (Jargon File) tartalmazza, amelynek neve kicsit félrevezető: a közel 10 MB-nyi anyag valójában a számítástechnika egyik leggazdagabb értelmező szótára.

Vegyes hálózat

A hálózatokkal kapcsolatos programok változatos összeállítása található a mostani CD-n. A NetWare és a közelmúltban megjelent Win2k együttélését szolgáló csomag NetWare szerverrel működő hálózaton dolgozó olvasóink figyelmébe ajánlható. A Windows munkaállomásokhoz készült csomag tartalmazza azt a NetWare klienst, amely a Windows 2000 programmal felszerelt kliensgép és a NetWare szerver együttműködését biztosítja. (A NetWare-hez kapcsolódó más Novell termékek és a Win2k viszonyáról lapunk 44. oldalán is olvashatnak.)

A két rendszer összehasonlításában nagy vitát váltott ki a címtárak együttműködésének kérdése. Olvasóink képet kaphatnak a Win2k címtáraként működő Active Directory v2.5 rendszeréhez kapcsolódó ADSI (Active Directory Service Interfaces) programozási lehetőségeiről. A közreadott csomag segítségével a COM komponenseket kezelni tudó valamennyi fejlesztőeszközzel olyan alkalmazások készíthetők, amelyek kihasználják a címtár lehetőségeit. A Microsoft Magyarország által rendelkezésünkre bocsátott készletben található ADSI25 .CHM állomány általános help az ismerkedéshez, illetve az SDK (Software Development Kit) használatához. A különböző platformokon (Windows 95, 98, NT Intelen és Alphán) külön öntelepítő fájlok gondoskodnak az ADSI installálásáról.

A hálózatos programozás iránt érdeklődőknek szól az Inprise ajánlata. E havi CD-mellékletünkön a hálózatokon elosztott objektumok kezelésére alkalmas Visibroker For Java 4.0 próbaverzióját adjuk közre. A 60 napig használható demóverziók egyike Win32-re, a másik Red Hat Linuxra telepíthető, ilyenkor a „License Key” és „Company” mezők üresen hagyandók. Az installáláshoz és a használathoz kell a korábbi hónapokban CD-nkre felkerült JDK 1.2.x is.

Vendégoldal

A Borland szabaddá tette a 32 bites Windows platformra készült C++ fordító 5.5-ös, parancssoros verzióját. Ugyancsak Borland eszköz az InterBase, melynek 6.0-s linuxos béta verziója került fel a CD-mellékletre.

A Microsoft kínálatából a hálózatos címtárhasználat programozását segítő ADSI mellett közreadjuk az Office 97 második javítócsomagjának CD-ről is telepíthető verzióját (angol és magyar nyelvűhöz is). Fontos, hogy a 2. előtt az 1. javítócsomag legyen telepítve!

A Samba a Unix rendszerek SMB (Server Message Block) prtokollon alapuló fájlelérési eszköze, amely a Linux terjedésével valószínűleg egyre népszerűbb lesz, ezért is tettük fel mellékletünkre annak tankönyvével együtt.

Simay Endre István

Miért nincs szoftverügynökség?

Régebben is előfordult már, hogy kedvem támadt volna regisztráltatni egy-egy jól bevált shareware programot, legutóbb azonban az Audiotools kapcsán a gondolatot tett is követte. Nagylemezeimről akartam magamnak MP3 fájlokat készíteni, és erre az Audiotools kiváló eszköz. A baj csak az, hogy a shareware változatba beépített 5 perces (illetve az új verziónál 10 perces) időkorlát miatt egyetlen oldalnyit sem lehet végig felvenni vele. A teljes változat pedig csak a program készítőjétől szerezhető be. Nos, a hazai viszonyok között ez nem is olyan egyszerű.

A tapasztaltabbak külön bankszámla nyitását javasolták, de én is féltettem az alapkártyámat az internettől. A szerző honlapján persze láttam, hogy csekkel is lehetne fizetni (és 20 dollár az egész), viszont a banknál a csekk elintézése ugyanakkora macerával, adminisztrációval járt volna (kérdőív stb.), mintha több kamionnyi árut hozattam volna be... Végül tehát döntöttem a „biztonsági" bankkártya kiváltása mellett, vállalva a többhetes várakozást, a tulajdonképpen felesleges második számlát, meg annak költségeit, ami évente mégiscsak kitesz néhány ezer forintot.

A kérdésem mindössze annyi, hogy okvetlenül így kell ennek lennie? Tényleg nem tudunk kialakítani valami egyszerűbb, olcsóbb és gyorsabb megoldást a Magyarországon történő regisztráláshoz? (Azoknak, akik a megkrekkelt változatok helyett inkább ezt választják.) A sok bálás butik mellett elkelne az országban egy nem túl mohó szoftverügynökség, amely akár az ASP égisze alatt közvetítő szerepet töltene be a shareware programok készítői és a regisztrálást igénylő felhasználók között. Erre vajon miért nincs vállalkozó? Szívesen fogadnám, ha valaki tudna (és hajlandó lenne) erre válaszolni.

Czeglédi Nagy Miklós
miku@mail.uti.hu

A szoftver minősége

Visszatérve még mindig a januári hónap témájához, álmom egy olyan szoftver, amelyből nem kell folyton új verzió, mert már az elején becsületesen, lelkiismeretesen elkészítették. Ha mégis szükséges a frissítés, akkor minden további (konverzió) nélkül kezelje a korábbiakkal készült dokumentumokat, ne legyenek a menüpontok öncélúan átszervezve, ismerje fel minden telepített karakterkészlettel a hosszú „ő" és „ű" betűt, ne méretezzen egy szóról szóra megegyező dokumentumot kétszer akkorára, mint a korábbi változat stb.

Még lehetne folytatni, de szinte képtelenség mindent leírni. És minél hosszabb a felgyülemlő bűnlajstrom, annál inkább a guta kerülget a gátlástalanul öntömjénező süket reklámrizsa láttán. Tessék mondani, ha valaki ilyen trehánysággal építene hidat, toronyházat, meddig maradhatna szabadlábon? (Természetesen a hivatkozott szoftverekben vannak jó dolgok is. Feltehetően akadt néhány lelkiismeretes beszállító...)

A múlt számban általam említetteken kívül szívesen olvasnék a lapban olyan összeállítást is, ahol az egymással megférő hardverelemek (alaplap, processzor, memória, hang- és videokártya) típusonként csoportba lennének szedve. Olvasom a különböző ezközök teszteredményeit, halvány utalások esetleg vannak, hogy valamely eszköz egy másikkal megy, egy harmadikkal pedig össze sem építhető, sőt még a doboz mérete is gondot okoz. Ha bemegyek egy boltba, akkor azt a variációt akarják rám erőltetni, amelyik éppen kapható, hiába lenne nekem árban és tudásban jobb az, amit a megszerzett információk alapján elképzeltem. Egy kompatibilitási listára támaszkodva esetleg jobban össze tudnám válogatni, akár különböző boltokból egyenként „összevadászva" mindazt, ami nekem megfelelő.

Tiberius Péter
tiberius@hcbyte.hu

Ha nem jön a hang

Olvastam Smaragd László kérdését az Új Alaplap legutóbbi számában, az 57. oldalon. Nos, nekem is volt hasonló problémám, amíg Windowst használtam. A 486 DX100-as processzorú számítógép ESS 1868-as hangkártyával és Win95-tel volt „felszerelve". A Windows szabályosan felismerte a hangkártyát, telepítette a drivereket, és minden működött volna, ha... Ha a Microsoft tökéletesen írta volna meg azt az operációs rendszernek nevezett valamit.

Beszereztem hozzá az újabb kiadású meghajtóprogramokat is, de változatlanul süket maradt a kártya. Ekkor az eszközkezelőben, a hangkártya tulajdonságlapján átállítgattam a megszakításokat, a DMA-t úgy, hogy ne legyen ütközés más perifériákkal. Ezt oda-vissza megcsináltam néhányszor (az sem volt baj, hogy az eredeti állt vissza), és lássanak csodát, a kártya megszólalt. Való igaz, hogy ez a „megoldás" csak addig működött, amíg a gépet ki nem kapcsoltam, mert a következő alkalommal az egész procedura kezdődött elölről... A dolog pikantériája, hogy ugyanez a kártya Windows NT 4.0 és Linux alatt kifogástalanul szólt.

Sziráki Tibor
szikar@mail.datanet.hu

Hardverhiba? Driverhiba?

A hangkártyáról múltkor megírtakhoz képest vannak újabb fejlemények is. Próbaképpen átraktam saját hangkártyámat a szóban forgó gépbe (ez Genius, az egyik legolcsóbb típus), a BIOS-ban pedig letiltottam az audio controllert. Több mint egy órai küzdelem után sikerült szóra bírnom (ütközések, egyebek), így most nincs probléma a hanggal, és azoknak a játékoknak a kísérőhangja is szól, amelyek eddig némák voltak.

Persze ez még nem ad megnyugtató választ a kérdésre, mert lehetett hardverhiba, de a driverek is okozhatták. Ez utóbbit azonban nem hiszem, mert különböző illesztőkkel ugyanazt a jelenséget produkálta. Amennyire én tudom, a .wav és a .mid hangfájlokhoz más-más illesztőprogram szükséges, és itt mindkettő egyszerre akadt ki, ami még inkább megerősíti a hardverhiba lehetőségét.

Most már csak az a kérdés, hogy érdemes-e visszavinni az alaplapot a forgalmazóhoz, hogy cserélje ki. Azért még ma sem olyan nagyon egyszerű végrehajtani egy alaplapcserét, főleg ha valaki sohasem csinált ilyet. És mi van akkor, ha ez amolyan „szériahiba", tehát az ilyen típusú összes alaplapra jellemző? Vagy az is lehet, hogy még kiforratlan a technológia... De ha mégis driverhiba, akkor az üzletben sem tudnak segíteni.

Egyelőre azt javasoltam a gép tulajdonosának, hogy vegyen inkább egy hangkártyát, és ezzel megoldódik a probléma. De mint a számítógépek iránt komolyan érdeklődő ember, tudni szeretném, nincs-e mégis valami „gyógymód". Ha megjelenik a lapban a kérdésem, és lesz aki válaszol rá, azzal rajtam kívül talán másoknak is segítenek, akik hasonló problémába ütköztek.

Smaragd László
smaragd3@freemail.hu

Windowstól a nyílt forráskódig

Parancs a kék halálba

Furcsa hibára bukkantak a Win95 és Win98 rendszerben. A DOS prompton bizonyos kvázi parancsokat kiadva rendszerösszeomlás („kék halál" vagy lefagyás) következik be. Az egyik ilyen hatásos karaktersorozat:

```
c:\con\con
```

A rendszer megpróbálja végrehajtani a kiadott parancsot, de ezen a néven nem talál elindítható programot. Normális esetben ilyenkor egyszerű hibaüzenetet kapunk (Bad command or file name), ehelyett a rendszer megzavarodik, és rendszerint a Ctrl+Alt+Del sem segít. Komoly problémát önmagában ez nem okozna, de ilyen karaktersorozatok URL címekben is elhelyezhetők, és ha azokra a linkekre rákattintunk, ugyanúgy elszáll a gépünk, mintha a parancssorban ütnénk Entert a fenti karaktersorozatra.

Hasonló a helyzet, ha olyan levelezőprogramot használunk, amely hajlandó a benne lévő URL hivatkozásokat a hozzárendelt böngészőben automatikusan megnyitni. Veszélyeztetve vannak továbbá azok, aki webes levelezőrendszert, mondjuk Hotmailt használnak. Ezeknél a levél valójában a levelezést biztosító szerveren található honlapnak tekinthető. Egy levél megnyitásakor engedély nélkül aktivizálódhatnak a beágyazott URL címek, bennük a „rendszergyilkos" karaktersorozatokkal. A fő veszélyforrás az, hogy ilyen trükkel a számítógépek távirányítva újraindíthatók (a felhasználó előtt ugyanis nincs más kiút), ami kiindulópontja lehet valamilyen cracker tevékenységnek. Például a Pentium III beépített azonosítószámát szokták a tulajdonos tudta és beleegyezése nélkül visszakapcsolni, de egy trójai program is aktivizálható.

A hiba adatvesztést (elvileg) nem okoz, de az előfordulhat, hogy böngészés közben egy nyitva maradt Word ablakban a feledékenységből el nem mentett munka odavész. Az egésznek az az oka, hogy a DOS meghajtónevei (COM1, CON, LPT1 stb.) a Windowsban fájlnévként és könyvtárnévként nem használhatók. Ha valaki mégis ilyen kombinációkat ad meg, akkor a rendszer összeomlik. A Microsoft gyorsan elkészítette és honlapjáról letölthetővé tette a javításokat. Windows 95-re: http://www.microsoft.com/downloads/release.asp?releaseID=19491 , Win98-ra ugyanez, csak az utolsó 5 számjegy: 19389. *(CNet)*

Word és Excel a zsebben

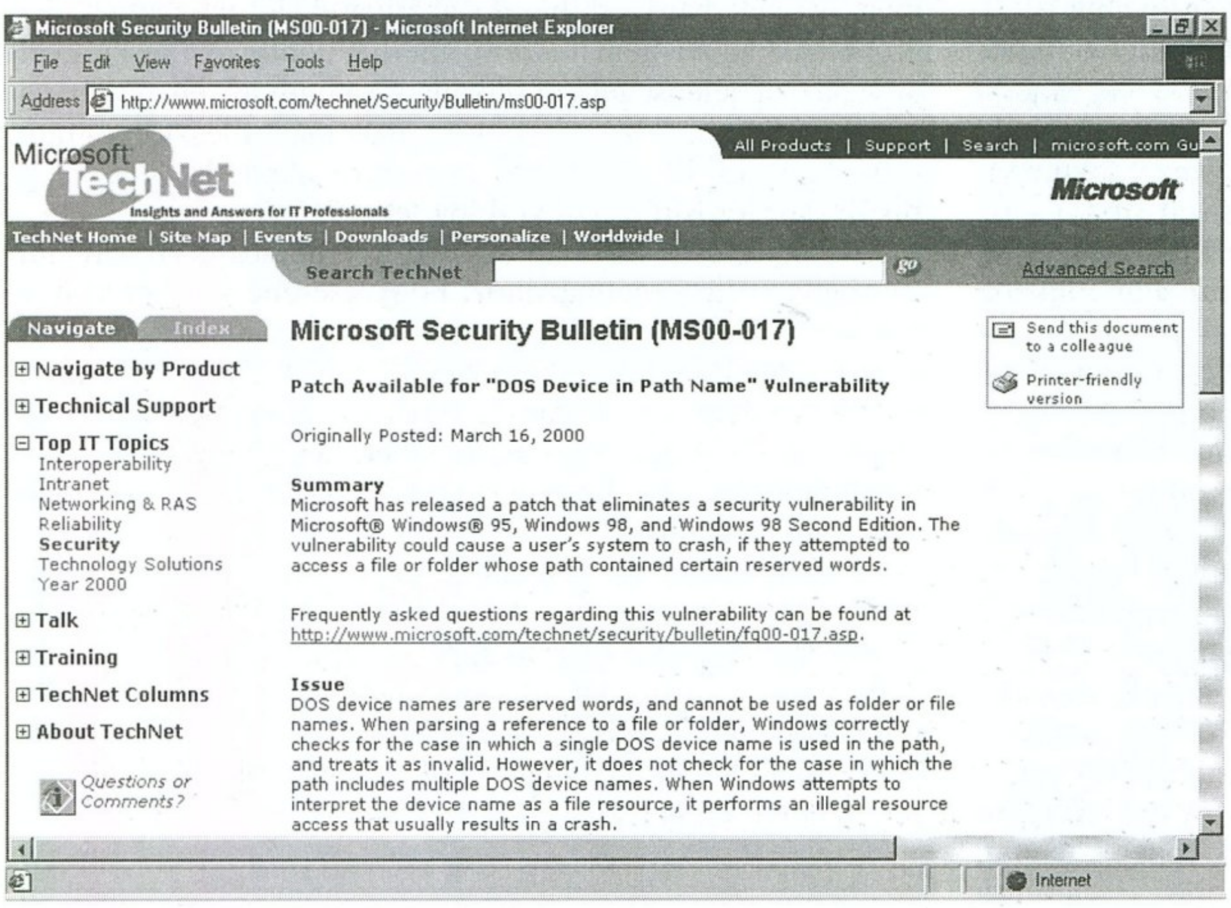
A Microsoft tervei szerint az új PocketPC jóval felülmúlja az eddigi palm kategóriájú gépeket, mert az Excel és a Word ezen a platformon is futtatható lesz. A Microsoft akkor döntött erről, amikor egyértelműen kiderült, hogy a palm gépeken használatos Windows CE visszaszorult, és sokkal népszerűbb lett a Palm OS. Redmondban ezért új hardver és új stratégia mellett döntöttek. A PocketPC más irányvonalat képvisel, mint a korábbi palm gépek, mert egyfajta utazó irodának fogható fel. Elkészül hozzá a Pocket Outlook levelezőprogram újabb változata, és erre a platformra is átírják a Microsoft pénzügyeket kezelő Money programját. A PocketPC-n lehetőség lesz MP3 és Windows Media Player formátumú állományok lejátszására, sőt elektronikus könyvolvasásra is. Ez utóbbihoz a Microsoft új programot fejlesztett ki, amely a hírek szerint jól olvashatóan jelenteti meg a szövegeket kis képernyőn is.

A Word és Excel formátumú állományok már korábban is használhatók voltak a Handheld PC és a Handheld PC Professional változatokon, előbb azonban az asztali számítógépen egy program segítségével konvertálni kellett őket. A PocketPC már képes lesz ilyen állományok közvetlen olvasására, szerkesztésére és természetesen elmentésére. A PocketPC megjelenése május végére, június elejére várható, és azt a Compaq készíti. *(ZDNet)*

Webszörf útlevél gyerekeknek

A Microsoft egyik új terméke és a hozzá kapcsolódó akciója révén elvileg visszaszorítható lesz az a jelenség, hogy a weben szörföző gyerekek az interaktív (főleg elektronikus kereskedelmi és szolgáltatási) weblapokon olyan információkat adjanak meg önmagukról és szüleikről, ami később jogi bonyadalmakhoz vezet. A törvény értelmében ugyanis a kiskorúaktól információt beszerezni csak szülői egyetértéssel lehet. Bár a honlapokon legtöbbször feltüntetik, hogy mi történik a rendelkezésre bocsátott név, cím, telefonszám stb. adatokkal (például, hogy továbbadják-e azokat), a gyerekek ezzel nem sokat törődnek.

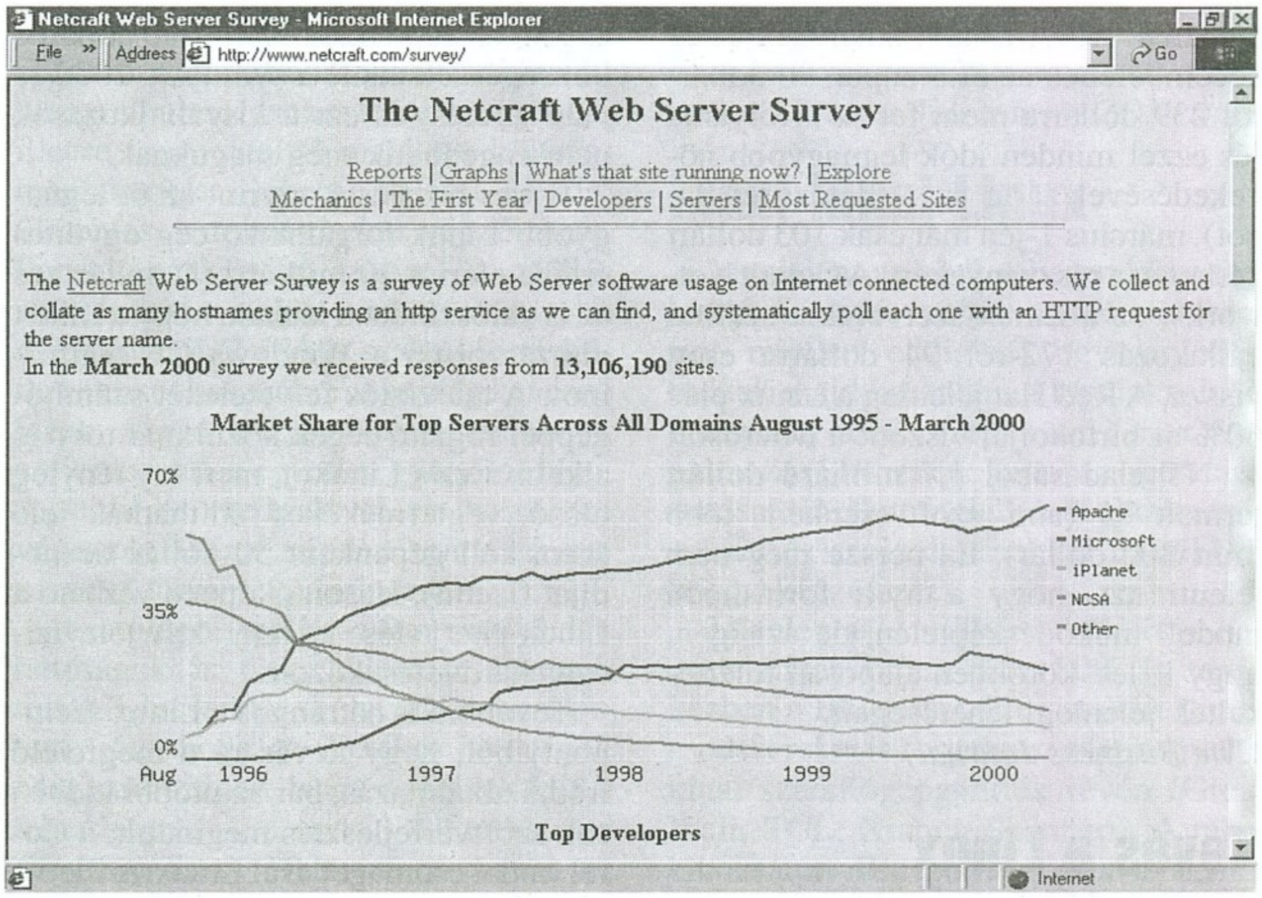
A látogatóktól adatokat kérő cégeknek maguknak kell tehát vállalniuk, hogy webszerverükre felteszik a Micro-



soft Passport programot, a szülők pedig gyermeküknek regisztráltatnak egy webes „gyermekútlevelet” (Kids Passport), meghatározott jogosultságokkal (e-mail cím megadása stb.).

Amikor a kiskorú olyan weblapra lép be, amely részt vesz ebben az önkorlátozásban, akkor a szerver az „útlevél” adataiból megállapítja a gyerek életkorát, és megjelenít egy üzenetet, hogy a továbblépéshez szülői beleegyezés szükséges. A gyerek szól valamelyik szülőjének, aki a a számítógép előtt megadhatja csemetéjének az engedélyt. A szülői (vagy felnőtti) illetékesség igazolásra egy érvényes hitelkártyaszám szolgál. Ez így elméletileg nagyon szép, de a gyakorlatban számos probléma teheti működésképtelenné az egészet. Az utóbbi időben például rengeteg ügyfél hitelkártyaszámát szerezték meg különféle szolgáltatóktól, a szülői felügyelet ilyetén gyakorlása ezért egyáltalán nem kockázatmentes.

(PC WorldOnline)



A szerverek tükrében

A Netcraft 2000. márciusában 13 106 190 webhely adatai alapján készítette el szerverekre vonatkozó szokásos felmérését. Egy hónap alatt az Apache egymillió új szerverre került fel és aránya meghaladta a 60 százalékot. A többiek arányvesztése pedig tovább folytatódik (lásd a mellékelt képen).

A legnagyobb fejlesztők szerverszoftvereinek arányait mutatja az alábbit táblázat. Abban az „iPlanet” közös név jelenti az iPlanet Enterprise, a Netscape Enterprise, a Netscape FastTrack, a Netscape Commerce, a Netscape Communications, a Netsite Commerce és a Netsite Communications programokat. Mindezek mögött a Sun–Netscape szövetség áll. A „Microsoft” név a Microsoft Internet Information Server, a Microsoft IIS-W, a Microsoft PWS-95 és a Microsoft PWS szervereket jelöli. A Netscape már elkezdte szervereinek átállítását, így azok aláírásai a jövőben az „iPlanet WebServer Enterprise 4.1” üzenetet fogják tartalmazni a „Netscape Enterprise 4.0”. helyett.

A Netscape-nek szerver szempontból 1996 volt a legsikeresebb éve, de azóta inkább az „electronic commerce server” kategória felé mozdult el. Ami az egyes szoftvereket illeti, csak az Apache tudta az arányát növelni, az összes többié csökkent . Az iPlanet-et a vezető brókercégek részesítik előnyben, viszont az internetes kiskereskedelemben a Microsoft már több mint 50%-os arányt ért el (a stores.org és a Media Metrix top 100-as listája szerint).

Érdemes megjegyezni, hogy a vertikális modell (vagyis nem csupán a szerverszoftver, hanem a teljes rendszer és a komplex szolgáltatás) egyre jobban terjed, és ezzel együtt az Apache pozíciói is erősödnek, mert az internetes ipar nagy része az Apache-ot választja. A szigorúan vertikálisan építkező bankok azonban az iPlanet felől inkább a Microsoft irányába mozdulnak el. Ami az egyes mammutcégek preferenciáit illeti, szintén érdekes lehet, hogy a Compaq Solarist és Apache-ot használ a számításba jöhető egyéb rendszerek (NT, Linux, FreeBSD, Digital Unix, Guardian stb.) helyett.

Az utóbbi időben kétszer is feltörték az RSA szerverét, ezért mindkét eset után lecserélték a rendszert. (Mellesleg az RSA „a legmegbízhatóbb név az e-security-ben” szlogennel reklámozza magát.) Eközben a Windows 2000 a megjelenése után, február közepén elért mintegy 20 ezer webhelyről egy hónap alatt 40 ezerre ugrott. Az elemzők szerint, ha a Microsoft rá tudja venni a nagy internetes cégeket, hogy térjenek át a Windows 2000 rendszerre, akkor a koncentráltság miatt drámai változások játszódhatnak le a szerverpiacon, mert jelenleg éppen ezek a cégek a Linux legnagyobb támogatói és használói. (http://www.netcraft.com/survey/)

A szerverszoftverek fejlesztőinek élmezőnye

	Apache	Microsoft	iPlanet
2000 január (db)	6 482 708	2 447 629	867 264
Részarány (%)	58,08	21,93	7,77
2000 február (db)	7 870 864	2 742 931	955 148
Részarány (%)	60,05	20,93	7,29
Arányváltozás (%)	+1,97	−1,00	−0,48

Vége a tőzsdei Linux-őrületnek?

A jelek szerint a befektetők számára nem igazán vonzó egy olyan konstrukció, ahol az 5,7 milliárd dolláros szerverpiac 25%-ának birtoklása a bevételnek 1%-át sem éri el. Ez kihatott a tőzsdei árfolyamra, és a linuxos részvények értéke meredeken visszaesett. A Red Hat a korábbi 151 dolláros árfolyamhoz képest 55 dolláron zárt 2000 március 1-jén. A tőzsdeanalízissel foglalkozó Tomi Kilgore egyenesen úgy fogalmazott, hogy a Linuxnak már nem szentelnek akkora figyelmet, mint korábban, és „az őrületnek nagyjából vége”.

Az IDC felmérése szerint a Linux a szerverpiacon az 1998-as 17%-ról 1999 végéig 25%-ra növelte részesedését, és egyedül a Windows NT előzi meg (38%). Ugyanakkor az 1,35 millió Linuxért 1999-ben alig 36 millió dollárt fizettek ki (vagyis mindent egybevetve darabonként 27 dollárt), ami kevesebb az 5,7 milliárd dolláros szerverpiac

1%-ánál. A VA Linux, amelynek 1999 decemberében az első napon 30 dollárról 239 dollárra ment fel az árfolyama (és ezzel minden idők legnagyobb növekedésével zárta az első tőzsdei napot), március 1-jén már csak 103 dollárt kérhetett a részvényekért. A Cobalt Networks, ez a Linux-szervereket szállító vállakozás 172-ről 94 dollárra esett vissza. A Red Hat jelenleg a Linux piac 60%-át birtokolja, viszont a Microsoft az NT eladásából 1,7 milliárd dollárt termelt ki (ami szoftverenként több mint 800 dollár). Ez persze még nem jelenti azt, hogy a nyílt forráskódú modell működésképtelen, de azt igen, hogy üzleti körökben alaposan túlértékelték jelenlegi lehetőségeit.
(The Business Journal)

Kevés a Linux az irodákban

Egy ausztrál felmérés szerint a Linux már nem egy marginális operációs rendszer, hanem az NT komoly vetélytársa. A Computer Daily News 417 ausztrál céget kérdezett meg, közülük 43% használ rendszeresen Linuxot, további 24% pedig teszteli azt. Csak 29% nyilatkozott úgy, hogy a jövőben biztosan nem fogja használni. A desktop-kategóriában a „pingvines" oprendszernek nem sokan jósolnak nagy jövőt, mindössze 13% lát erre esélyt.

Költség oldalról a Linux mellett szól, hogy még akkor is legfeljebb 80 dollárba kerül, ha keménydobozos verzióját vásárolják meg (de az internetről ingyen is letölthető), míg a Giga Information Group előrejelzése szerint a cégeknek gépenként 970–1640 dolláros költséget jelentene a Windows 2000-re való áttérés, és ezt a kisvállalkozások nem engedhetik meg maguknak.

Egyes becslések szerint az öt legnagyobb Linux-forgalmazó cég együttes értéke eléri a 20 milliárd (!) dollárt, és az is valószínűnek látszik, hogy a Linux visszaszorítja a Windows CE platformot. A televíziós felvételeket számítógéppel rögzítő cégek közül már több is alkalmazza a Linuxot, mert így tényleg olcsó végtermékeket állíthatnak elő (nem kell gépenként 50 dollár licencdíjat fizetni). Hasonlóképpen várható a Linux elterjedése az interaktív televíziózás hardvereszközein.

Továbbra is hátrányos a Linux szempontjából, hogy kevés rá a megfelelő irodai alkalmazás, bár az utóbbi időben sok szoftverfejlesztés megindult. a Corel irodai csomagot árul Linuxra, illetve van WordPerfect Linuxra, és készül a CorelDraw is (amely várhatóan ugyanannyiba fog kerülni, mint a windowsos verzió). A Sun céges használatra is szabaddá tette Star Division-től megvett StarOffice-t, és az Applix ugyancsak forgalmaz linuxos irodai programcsomagokat. Ugyanekkor a Windows 2000-re nem kevesebb mint 7000 alkalmazás van, nem meglepő tehát, hogy a világ irodai desktop számítógépeinek alig 6%-án fut Linux. *(AFR — USA Today — PC World — NewsAlert)*

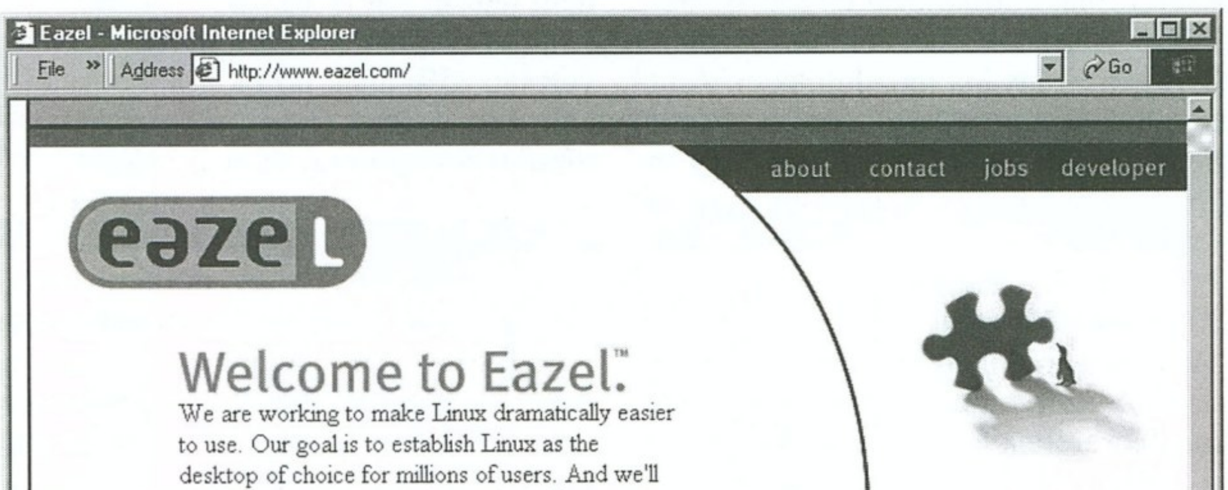
Desktopon az áttörés késik

Bár a legutóbbi CeBIT-en a Linux roppant népszerű volt, még leglelkesebb támogatói szerint is várni kell arra, hogy a desktop kategóriában igazán sikeres legyen, mert az irodai használatban még sok mindent kell pótolni. Roland Dyroff, a SuSE Linux vezetője szerint a másik hiányosság a nem megfelelő hardvertámogatás, bár az is igaz, hogy leginkább azoknak lehetnek gondjaik, akik „még nem láttak Unixot". A Linux installálásának tényleges nehézségi szintjét a kívülállók azért sem tudják reálisan megítélni, mert a Windows az esetek nagy részében már telepítve van a megvásárolt számítógépen, de a preinstallált Linux elterjedésével ez a helyzet meg fog változni. A SuSE 6.4-es verziója várhatóan áprilisban fog megjelenni, és nem csupán új grafikus installálási felülete lesz, hanem a desktop is megújul. Adnak hozzá továbbá angol és német nyelvű kézikönyvet is. A SuSe idén Ázsia és Dél-Amerika felé is nyitni akar. *(IDG)*

A Linux GUI fejlesztői

Két új grafikus felhasználói felületet is fejlesztenek Linuxra. Az Apple és az AOL szakemberei közösen dolgoznak azon, hogy a Macintoshéhoz hasonlóan könnyen áttekinthető és könnyen kezelhető felületet hozzanak létre. Az Eazel (http://www.eazel.com) szerintük azért szükséges, mert a jelenleg legelterjedtebb grafikus megoldások (amilyen a Gnome és a KDE) még messze vannak a kiforrottságtól, és nem is igazán felhasználóbarátok. Néha nehéz elkerülni, hogy parancsokat kelljen begépelni, és ettől sokan idegenkednek, sőt néha félnek. Az Eazel a Gnome-on alapul, a project vezetője, Mike Boich 1982-ben csatlakozott az Apple-hez, és részt vett a Macintosh fejlesztésében. A „szoftvervarázsló" Andy Hertzfeld három évvel korábban került az Apple-hez, hogy perifériákat és szoftvereket fejlesszen az Apple II-re, majd pedig a Macintosh csapat alapító tagja lett. Az AOL egyik igazgatóhelyettese, Mike Homer szintén az Apple II-vel kezdte fejlesztői pályáját 1982-ben.

Az Eazel mellett természetesen másutt is dolgoznak a felhasználóbarát, grafikus felhasználói felületű Linux kialakításán. Megemlíthető a MandrakeSoft vagy a CorelLinux, ám az Eazel azt tervezi, hogy az új, nyílt forráskódú szoftver egyszerre fog fájlmmenedzserként és grafikus héjként funkcionálni, kezelhetőségének könnyedségében és rugalmasságában pedig túltesz majd a Windowson és a Macintoshon. Emellett a Linux komplexitását sem akarják feláldozni.



A Gnome-ot fejleszti tovább Miguel de Icaza is, akinek a nevéhez ez a grafikus felhasználói felület fűződik. Az 1999-ben általa alapított HelixCode (http://www.helixcode.com) által kibocsátott Helix Gnome desktop béta verziója már kevésbé bonyolult, és többek között automatikus installáló és szoftverfrissítő is segíti majd a felhasználókat. Az új verziót minden eddiginél könnyebb lesz a Red Hat, a Mandrake, a SuSE, a Caldera, a LinuxPPC disztribúcióira installálni. A HelixCode a Microsoft Office csomag nyílt forráskódú, Linuxos megfelelőjét is fejleszti. Ebben megtalálható a Gnumeric nevű táblázatkezelő és az Evolution kommunikációs csoportmunka csomag (beépített e-mail klienssel, naptárral és címjegyzékkel), mert az reméli, hogy így a nyílt forráskódú operációs rendszer vonzóbb lesz a felhasználók számára. *(CNET News.com — The Standard — Wired News)*

Közbeszerzési szempontok — ott...

A norvég hatóságok mérlegelik, hogy nem volna-e érdemes áttérni a Windowsról a Linux platformra. Egy most készülő felmérés azt vizsgálja, hogy ehhez a minisztériumoknak milyen linuxos alkalmazásokra lenne szükségük, illetve hogy ezáltal mekkora költségmegtakarítást lehetne elérni. Endre Groentes, a felméréseket végző Statskonsult szakértője szerint az átállás azért is fontos volna, mert az eleve több disztribúcióból álló és szabad forráskódú Linuxot használva az ország kevésbé lenne függő helyzetben, mint most, amikor minden egyetlen gyártótól függ. A felmérés eredményei ez év végére várhatók, utána minden minisztérium maga eldönti, hogy mit tesz.

A norvégok mellett a francia és a dán hatóságok is érdeklődnek a Linuxra történő lecserélés iránt. (Nem hallottunk róla, hogy az illetékes hatóságok Magyarországon is mérlegelnék ezt a lehetőséget. — *A szerk.*) A StatMarket (WebSideStory) felmérése szerint az utóbbi 6 hónapban kis mértékben ugyan, de kimutathatóan visszaesett a Windowst használók száma a weben: jelenleg valamivel több mint 93% használja a Microsoft operációs rendszerét, a Unix 5%-ot, a Macintosh pedig 3%-ot tesz ki. Ez utóbbi kissé emelkedőben van az „egyéb" kategória tagjainak részesedésével együtt (Linux, SunOS és a microsoftos Web TV). A StatMarket felmérése a szörfölők által használt operációs rendszereket együttvéve számolta. *(France Telecom — ZDNet)*

Igazi kínai Linux

Megjelent az első, teljesen kínai fejlesztésű Linux, melyet az ígéretek szerint minden eddiginél egyszerűbben tudnak majd használni a kínaiaknak. A Bluepoint nevű vezető kínai Linux-forgalmazó (Shenzhen, Kína) az operációs rendszer mellett foglalkozik alkalmazásokkal és szolgáltatásokkal is (többek között képzéssel).

Jelenleg a 13 milliárd dollárosra becsült kínai Linux-piac 80%-át tartja kézben a Bluepoint. Preinstallált Linux rendszereket is forgalmaz a legnagyobb kínai számítógépgyártók révén (Great Wall, TCL, Xiahua Shanbao). A most kibocsátott Bluepoint Linux a kernel szintjén támogatja a kínai nyelvet, de a felhasználók teljes támogatást kapnak több más ázsiai nyelvhez is.

Az elemzők szerint a Bluepoint Linux a kínai Red Hat-nek tekinthető, honlapjai (http://www.openunix.org és http://www.bluepoint.com.cn) a leglátogatottabb linuxos webhelyek a világ legnagyobb országában. *(NewsAlert)*

Galántai Zoltán – Mákos András

INFORMÁCIÓKÉRÉS: 25 ▶

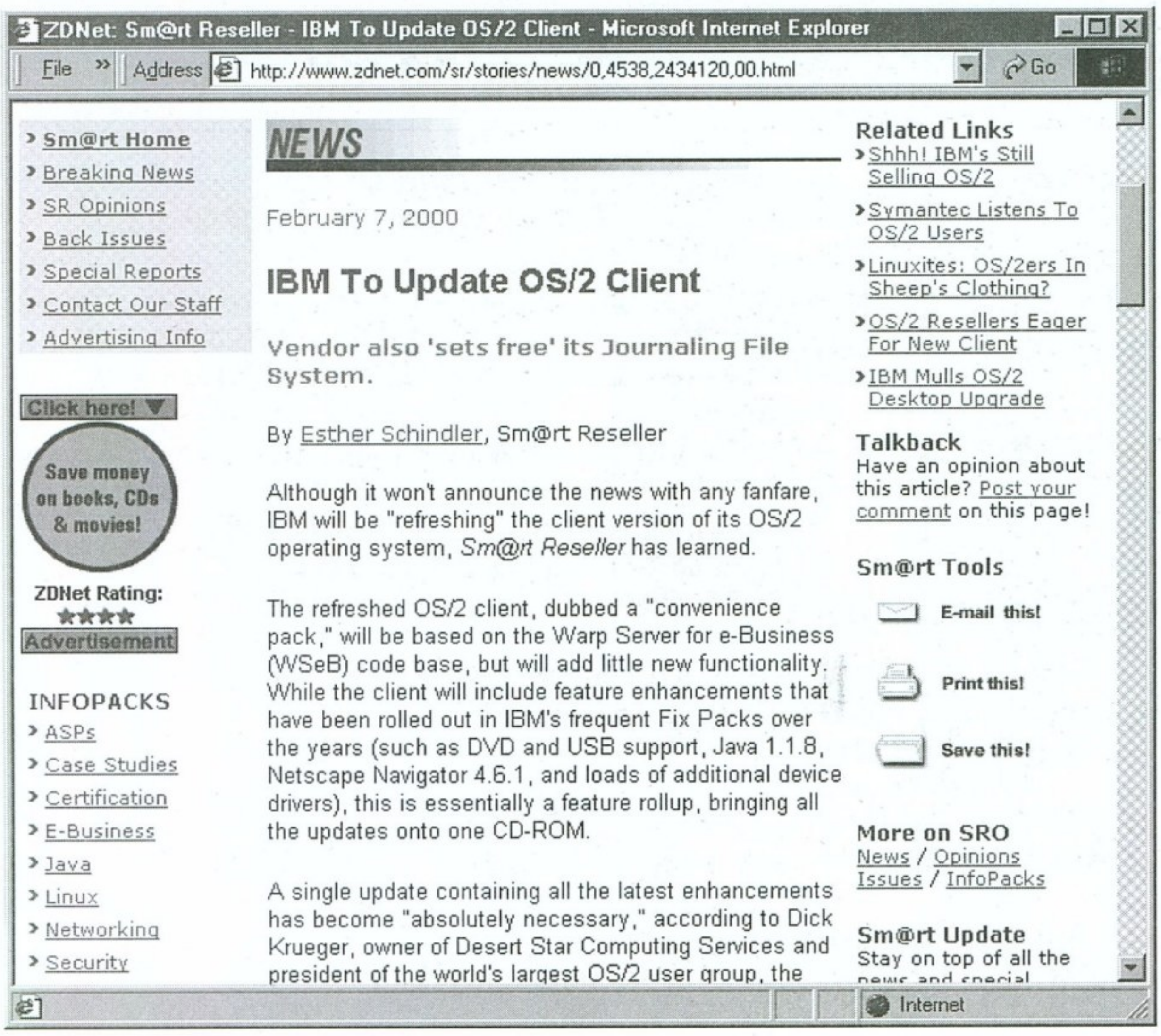
Nem csak OS/2 ...

Mégis lesz új OS/2 kliens?

Meg nem nevezett forrásokra hivatkozva röppentette fel a ZDNet a bomba hírt (http://www.zdnet.com/sr/stories/news/0,4538,2434120,00.html), miszerint az IBM mégiscsak kiadja az OS/2-es kliens továbbfejlesztett verzióját.

A tavaly megjelent Warp Server for e-Business alapjaira épülő termék nem fog túl sok új funkciót tartalmazni. Beépítik viszont a legújabb drivereket, javításokat, valamint mindazon bővítéseket, amelyek az elmúlt évek során jelentek meg az IBM Software Choice oldalán. A ZDNet forrása azt is tudni véli, hogy az IBM még legalább 7 évig továbbfejleszti az OS/2-t. A döntés állítólag az utóbbi évek meglepően kedvező eladási statisztikájának köszönhető.

Az IBM eddig még nem erősítette meg a hírt. Az interneten a cikk megjelenése után viszont rögtön megindult a spekulálás. Az OS2.ORG oldalán végzett közvélemény-kutatás szerint a felhasználók döntő többsége biztosra veszi a kliens megjelenését. Azt is tudni vélik, hogy az új verzió az év második negyedévében fog megjelenni, és Software Choice előfizetéssel lesz elérhető. Egyesek a kedvező fordulatot az IBM-en belül az utóbbi hetekben végrehajtott csendes átszervezésekben látják. Tény és való, hogy nemrégiben visszakerült az OS/2 fejlesztésével is foglalkozó részleg élére az a vezető személyiség, aki már az OS/2 első feltámadásában is közreműködött.



VMWare 2.0 béta

Az elsősorban PC-emulátorok fejlesztésével foglalkozó VMWare cég kibocsátotta a 2.0-s emulátor béta-változatát, amely most már az OS/2 futtatását is támogatja. A 30 napos próbaverzió segítségével az OS/2-nek az NT-n vagy Linuxon történő futtatása megvalósul. (A http://www.vmware.com/support/win/doc/beta_win/index_os2.html és a http://www.vmware.com/support/linux/doc/beta_linux/index_os2.html oldalakról tölthető le.) A felhasználók tapasztalata szerint a maga kategóriájában igencsak gyorsnak mondható emulátorban meglehetősen lassan fut az OS/2, ami a béta kódnak tesztelésre és nem pedig teljesítményre optimalizált voltával magyarázható.

A VMWare hivatalos tájékoztatása szerint még mindig nem döntötték el, hogy a 2.0 végleges kiadásában, vagy pedig csak az azt követő újabb verziókban válik-e hivatalossá az OS/2 támogatása.

Fájlrendszer a DVD-hez

Az IBM kiadta a DVD meghajtók OS/2 alatti teljes értékű használatához szükséges, telepíthető UDF (Universal Disk Format) fájlrendszert (UDF.IFS). A dolog szépséghibája, hogy az új szoftvert csak az előfizetéssel rendelkezők tölthetik le a Software Choice website-ról (http://service.boulder.ibm.com/asd-bin/doc/hu_hu/catalog.htm). Nem csoda tehát, hogy az interneten már a kiadást követő második napon megjelent a bővítés a kalóz ftp kiszolgálókon. A pletykák szerint a DVD-támogatást beépítik az új OS/2-es kliensbe is.

Nyilvános a JFS kódja

Valószínűleg az Journaling File System technológia (tranzakciós fájlrendszer) széles körű elterjedését reméli az IBM attól a döntéstől, amelynek keretében nyilvánossá tették az AIX és a Warp Server for e-Business operációs rendszerekben használt JFS fájlrendszer kódját. A GPL licenc keretében kiadott JFS a http://oss.software.ibm.com/developerworks/opensource/jfs/index.html oldalról tölthető le. A Linux körüli fejlesztők már rá is vetették magukat az elsősorban kiszolgálókban hasznos fájlrendszerre, és a hírek szerint rövidesen elkészülnek az átírással. Mivel a nyilvánossá tett kódban megtalálhatók az OS/2-specifikus részletek is, könnyen elképzelhető, hogy rövidesen megjelenik az ingyenes JFS-támogatás az OS/2-es kliensekhez.

RPM OS/2-re

Sok egyéb linuxos alkalmazás után az RPM-et (Red Hat Package Manager) szintén átvitték OS/2-re. Az RPM 3.0.3 segítségével az OS/2 felhasználók ezentúl kibonthatják gépükön a Linux CD-ROM-okon vagy ftp kiszolgálókon található csomagokat anélkül, hogy ehhez Linuxot kellene betölteniük. A dolog természetesen visszafelé is működik, vagyis ezentúl az OS/2-es alkalmazásokat is lehet RPM csomagként terjeszteni és telepíteni. Az átírást elvégző japán programozó, Sawataishi Jun je-

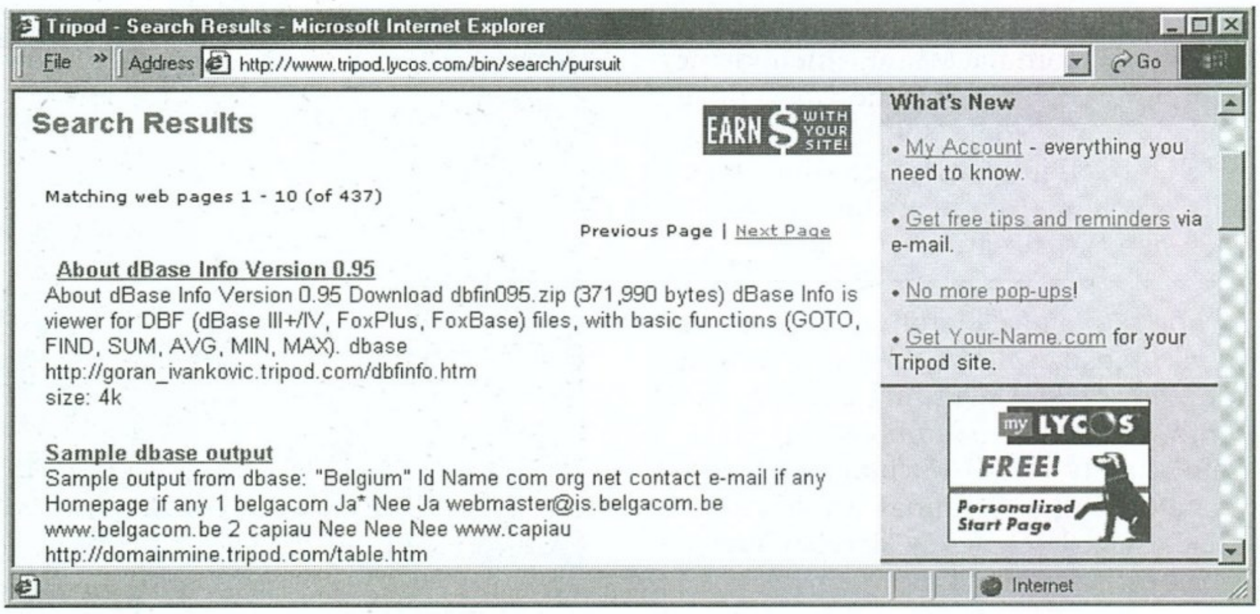
lenleg a DPKG (Debian csomagmenedzser) OS/2-es változatán dolgozik. A programok fellelhetők (lesznek) a http://www2s.biglobe.ne.jp/~vtgf3mpr/index-e.htm oldalon.

Jön a FixPak 13

Rövidesen kiadják a következő Warp 4-es javítócsomagot, a FixPak 13-at. Eddig is jelentek meg rendszeresen javítócsomagok, így ez a hír önmagában nem lenne különösebben érdekes. A pletykák szerint azonban a 13-as csomagban kitolják az OS/2-es VDM (Virtual DOS Machine, virtuális DOS környezet) memóriacímzési tartományát a jelenlegi 512 MB-ról 3 GB-ra, ami azt jelenti, hogy elvileg lehetséges lesz a 32 bites Windows alkalmazások futtatása a Win-OS/2 környezet frissítése után. A változtatást már tavaly bevezették a Warp Server for e-Businessben, és a beavatottak arról is tudnak, hogy az előző hírek egyikében említett új Warp kliens is a FixPak 13-on fog alapulni.

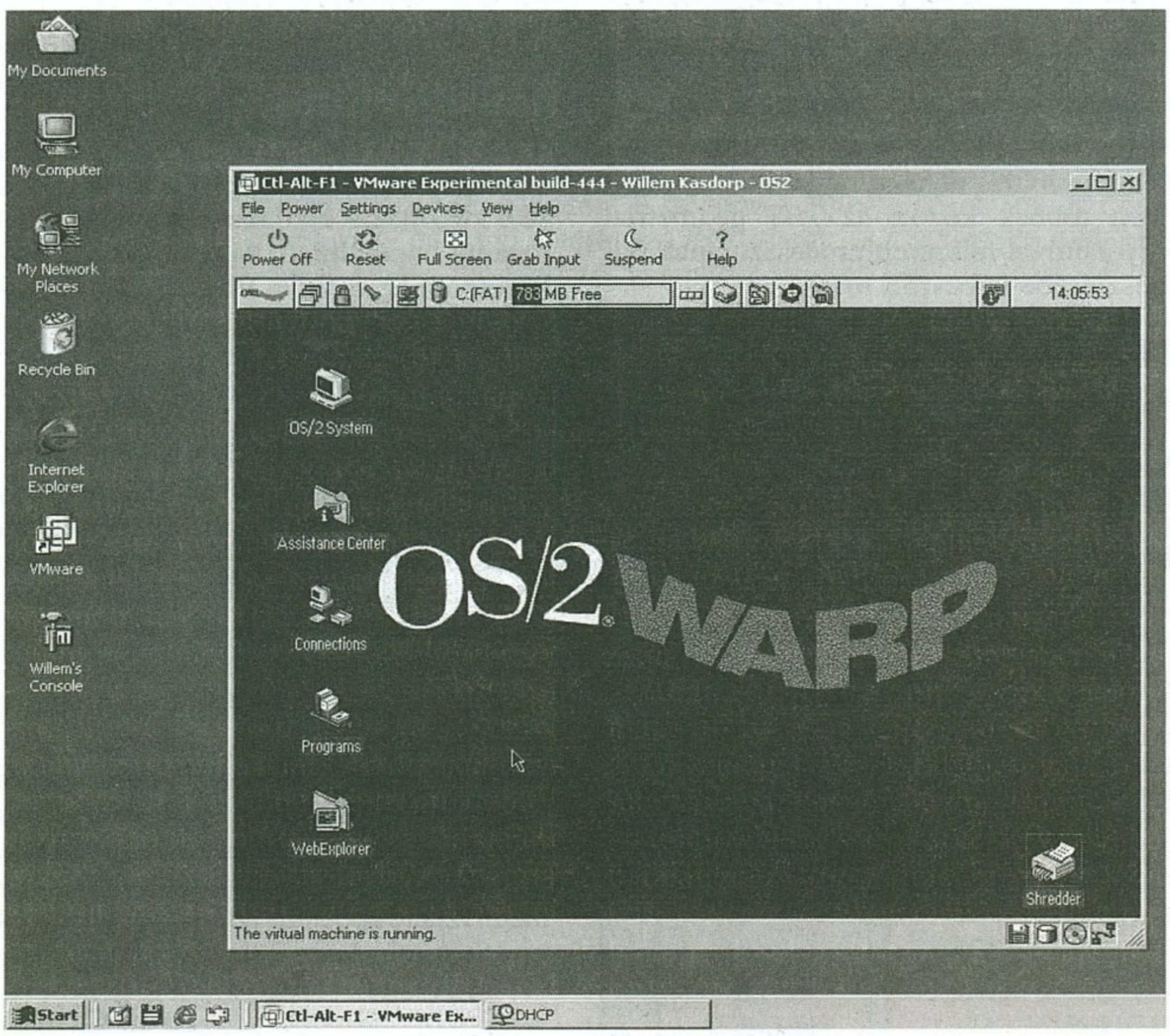
dBase Info 0.95

A dBase Info nem más, mint egy grafikus OS/2-es alkalmazás, amelynek segítségével DBF fájlokat lehet értelmezni és megjeleníteni. Az ingyenes (freeware) program egyelőre még csak az alapfunkciókat (GOTO, FIND, SUM, MAX, MIN, AVG) támogatja. Igény esetén a DBF fájlok SDF, ASCII, vagy akár HTML formátumúra is átalakíthatók. Külön érdekesség, hogy a programhoz a más nyelvekre történő átírást elősegítő eszközöket is mellékelték, így elvileg bármelyik lelkes honfitársunk elkészítheti a magyar nyelvű változatot. A dBase Info letöltéséhez a http://Goran_Ivankovic.tripod.com/ oldalra kell ellátogatni.



Megújult a Nice

Megjelent a Nice OS/2 Enhancer 3.1-es verziója. Az OS/2-es Munkaasztal csinosítására és funkcionalitásának kibővítésére használható ingyenes alkalmazás a http://hobbes.nmsu.edu/pub/os2/util/wps/nice-os2-v31.zip fájlban található. Ha valaki megunta az OS/2-es vagy Win-OS/2-es ablakok kinézetét, vagy éppen hiányolja a gyorsbillentyűk definiálásának lehetőségét, akkor feltétlenül próbálja ki ezt a programot!



Windows 2000 alatt továbbra is OS/2-ben

LPTool 2.11

A sokak által nagy megelégedéssel használt lemezmenedzselő programnak, a DFSee-nek az írója már régóta fejleszti az LPTool programot is. Az LPTool segítségével hálózati kártyával nem rendelkező gépek közötti adatcserét lehet megvalósítani a párhuzamos portokon keresztül történő összeköttetéssel. A 2.11-es verzió legjelentősebb újítása a tökéletesített könyvtárszinkronizációs mechanizmus, amelynek jóvoltából komplett könyvtárstruktúrák hozhatók szinkronba. Az ily módon létesített kapcsolat sebessége nem túl nagy (45 Kb/sec), ám megbízható. Az LPTool a fájlok másolásán kívül parancsok végrehajtását is lehetővé teszi a partnergépen. A DFSee és LPTool programok ingyenesen letölthetők a holland szerző, Jan van Wijk honlapjáról (http://www.fsys.demon.nl).

Links 0.82

Ha a böngészőket a méret és a funkcionalitás aránya alapján is osztályoznák, akkor a szöveges felületű Links valószínűleg nagyon jó jegyet kapna. A tömörítve mindössze 385 KB-ra rúgó alkalmazás igen gyors, korrektül kezeli a táblázatokat, és a gyorsítótár használata, valamint a megjelenítő karakterkészlete is konfigurálható. A cseh fejlesztésű program OS/2-es és Unix változatban a http://artax.karlin.mff.cuni.cz/~mikulas/links/ oldalon található meg.

Kádár Zsolt
kadzsol@xenia.sote.hu

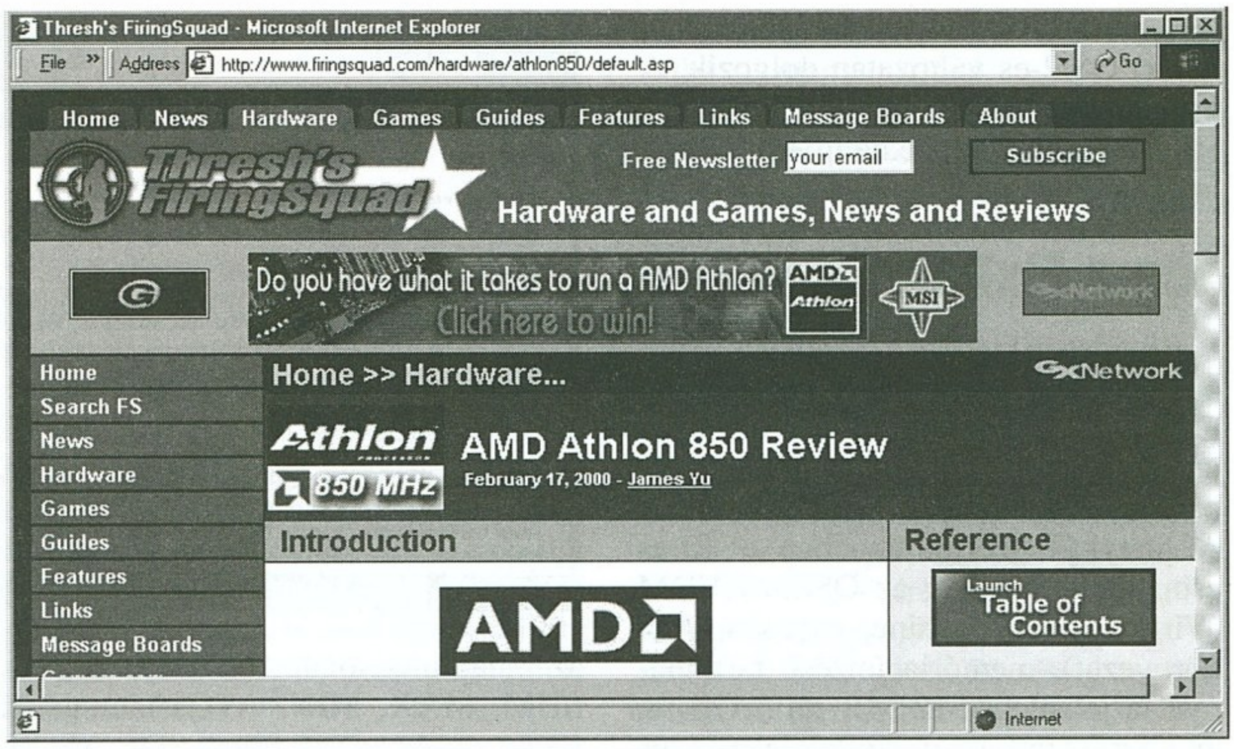
Elsőként a világon a Seagate kezd el 15000-es fordulatszámú merevlemezt gyártani. A 18 GB Seagate Cheetah X15 átlagos elérési ideje 3,9 ms, az Ultra160 SCSI modell július elején, a fibre channel ezt követően fog megjelenni, egyelőre még nem közölt áron.

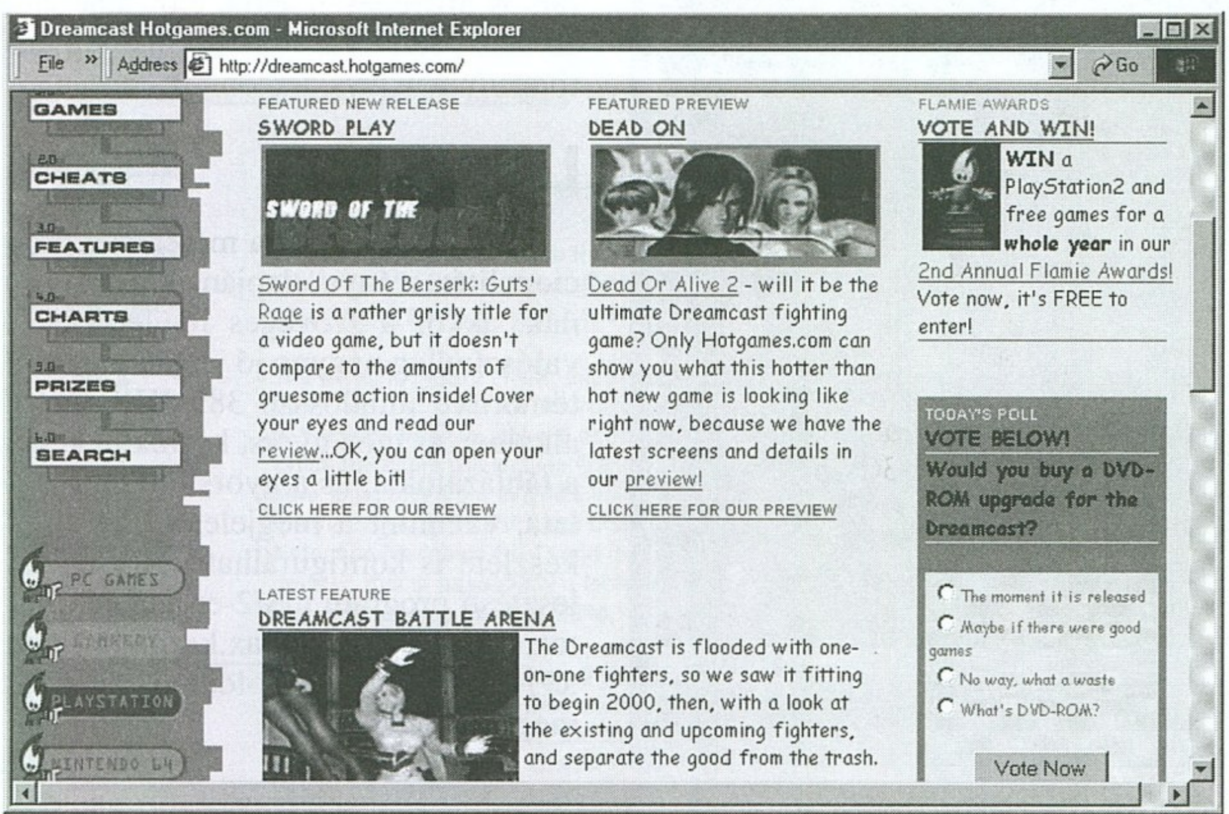
Folytatódik a Celeron sorozat. Az Intel egyik legsikerültebb processzorszériája mégsem szakad meg, március végén jelentik be az 566 és a 600 MHz-es változatot, amely már 0,18 mikronnal készül, és valószínűleg tartalmazni fogja az SSE utasításkészletet. Az év közepén már 700 MHz-es is lesz belőle.

Tovább folytatódik az árháború az Intel és az AMD között. Bár az AMD állítása szerint nincs összefüggés a konkurencia árcsökkentése és az általuk végrehajtott között, azonban a 800 MHz-es Athlon új ára (599 dollár) 47 dollárral van az azonos órajelű PIII alatt. A K7 750-es 489 dollár, a K6-2 500 MHz-es 93 dollár lett, ami 40%-os csökkenés. (A K6-2-ből pedig elkészült a leggyorsabb 550 MHz-es változat.)

A VIA megvette a Cyrix processzorok gyártási jogát, és hamarosan megjelenik a korábban Joshua kódnéven futó chipekkel. A Cyrix III kezdetben 400 és 433 MHz-en fut, de nagyobb sebességű rendszerbusz alkalmazásával az 500-533 MHz-es Celeronokkal fognak azonos teljesítményt nyújtani. Az Celeronnal kompatibilis, áprilisban megjelenő processzorok ára nagy tételben 89-99 dollár lesz.

Konkurenciaharc van kilátásban a drót nélküli otthoni hálózatok terén is. A jelenlegi Wireless Ethernet (802.11) mellett még a tavasszal megjelennek az első HomeRF szabványú eszközök, hat hónapon belül pedig lesznek gépek a kis távolságú kommunikációra alkalmas Bluetooth protokollhoz is (a távolság kb. 6-10 méter, átviteli sebessége 0,7-2 Mbps). Az otthoni hálózatok alkalmazása leginkább arra fog szolgálni, hogy az egy nagy sebességű internet-csatlakozást és a perifériákat megossza. A NEC már be is mutatta első olyan notebookját, amelyben a National Semiconductor chipsetje által vezérelt, beépített antennával rendelkező Bluetooth technológiájú adó-vevő van. Nyáron megkezdődhet a szállítása is (a vezérlő chip 40 dollárral megemeli az árát a normál notebookokhoz képest).





Az ATI 400 millió dollárért megvette az ArtX céget a Nintendo áldásával. A részvénycserével zajló üzletnek feltehetően nagy hatása lesz a cég új termékeire, mert az ArtX-nek jó üzleti kapcsolatai vannak a Matsushitával és az Acerrel, valamint a grafikus gyorsítók terén is nagy tapasztalattal rendelkezik. Az Silicon Graphics néhány tagja által alapított cég többek között az ATI geometriai gyorsítóval felszerelt processzorának fejlesztésébe is beszáll majd.

Speciális szemüveg nélkül is valódi térhatású megjelenítőt tervez az S3. A Drezdai Műszaki Egyetemen készített speciális D4D 3D monitorral és két S3 Diamond Fire GL1 grafikus gyorsítóval bármilyen Open-GL kompatibilis program képes lesz ilyen vizuális hatás elérésére. Az S3 mérnökei kifejlesztették a drivert, amely engedélyezi a két kártya ilyen használatát Windows NT grafikus munkaállomásokon.

Az európai mobiltelefonosok hamarosan új eszközt vehetnek a kezükbe. A Symbian legfrissebb terméke a Palm organizerhez vagy a kézi Windows CE gépekhez hasonló tudású Quartz Communicator 320x240 (negyed-VGA) felbontású színes képernyőt alkalmaz, elegendő teljesítménnyel a multimédia-lejátszáshoz, és támogatva az MPEG, az MP3 és a videokonferencia-szolgáltatásokat. A kézi készüléket csatlakoztatva mobiltelefonként használható, vagy a fent említett Bluetooth szabványú illesztővel hálózatba köthető. A cég részvényesei többek között a Nokia, az Ericsson, a Panasonic és a Motorola. Legérdekesebb a dologban az, hogy a tervek szerint „ingyen" lesz, ami azt jelenti, hogy a költségét az előfizetésen keresztül fogják a felhasználókra terhelni. (Kérdés, hogy milyen mértékben terjed el Magyarországon, illetve lesz-e olyan szolgáltató, amelyik vállalja a vele járó befektetést.)

A Segával közösen fejleszt karórát a Swatch. A Dreamcast konzol révén internetes információkat lehet betölteni az óra memóriachipjébe. Bár az óra önállóan nem képes megjeleníteni az adatokat, feltölthető másik Dreamcast gépbe, vagy megnézhető speciális adapterrel felszerelt kivetítőkkel. A megjelenést májusra tervezik, 130-180 dolláros áron.

Bánó György
gyuro@mail.vki.bke.hu

Váltson feljebb!

Windows NT Workstation 4.0 operációs rendszerét ingyen frissítjük Windows 2000-re!



Bár a **Windows 2000** magyar változata májusban jelenik meg, mégsem érdemes elhalasztania számítógépeinek beszerzését! Ha Ön 1999. november 1. és 2000. május 31. között vásárolt vagy vásárol Compaq Deskpro EP vagy EN számítógépet, a Compaq ingyen frissíti Microsoft NT Workstation 4.0 operációs rendszerét Windows 2000-re.

A Windows NT™ Workstation 4.0 utódaként a **Windows 2000 Professional** az üzleti célú desktop operációs rendszerek új generációját képviseli, amely megbízható számítástechnikai hátteret nyújt ahhoz, hogy cége megfelelhessen a gazdasági élet legújabb kihívásainak. A **Windows 2000** megnövelt rendszerstabilitásával, nagymértékű adatbiztonságával és az üzleti felhasználáshoz szükséges nagy teljesítményével garantálja vállalatának hatékony és zavartalan működését.

A Compaq mint a Microsoft legjelentősebb partnere kiemelkedő szerepet játszott a Windows 2000 létrehozásában – cégünk az új operációs rendszert eddig már több mint 18 000 belső PC-jére és 300 belső kiszolgálójára telepítette fel. A **Windows 2000** az eszközök, a szakértelem és a szolgáltatások terén ma egyet jelent a Compaq-kal: ma a Compaq kínálja a legszélesebb **Windows 2000** kompatibilis számítógépes választékot, és a Compaq termékek és szolgáltatások biztosítják a legjobb és legmegbízhatóbb **Windows 2000** platformot.

A Compaq mostani ajánlatával Ön hardver és szoftver tekintetében egyaránt a legújabb technológiát állíthatja cége fejlődésének szolgálatába. További információ a **www.compaq.hu** címen vagy zöld számunkon: **06-80-COMPAQ (266-727, 206-720).**

Az Intel, az Intel Inside embléma, a Pentium az Intel Corporation bejegyzett védjegyei, a Celeron az Intel Corporation védjegye.



Deskpro EP és EN sorozat

• Intel® Celeron™ processzor 466 MHz vagy Intel® Pentium® III processzor 550 MHz • 6,4 vagy 10 GB merevlemez • 64 MB memória (SDRAM) • Intel® Pro100+ WoL hálókártya • Matrox G200 AGP 2x videókártya 8 MB video RAM vagy Intel® videókártya integrált 4 MB video cache • AC ´97 sztereó hangkártya • Microsoft Windows NT 4.0 Workstation, magyar verzió.



INFORMÁCIÓKÉRÉS: 06 ▶

COMPAQ

Microsoft **Windows 2000** Professional

| Az új | gazdaság | alapja. |

intel inside pentium® III



INFORMÁCIÓKÉRÉS: 22 ▲



INFORMÁCIÓKÉRÉS: 13 ▲

Info 2000: május 9–13.

A Budapesti Vásárközpont „A" és „D" pavilonjában május 9–13. között kerül sorra az Info 2000 szakkiállítás. A legnagyobb tavaszi informatikai rendezvény tematikájába beletartozik az információtechnológia megannyi ága: a számítástechnika és a vele összefüggő elektronika, a telefónia és az adatkommunikáció, a reprográfia és az irodatechnika. Két hónappal a nyitás előtt a szervező Hungexponál mintegy 250 kiállító összesen 11 500 m^2 nettó kiállítási területet foglalt le. Az előző évi rendezvényhez képest jelentős az arányeltolódás a telekommunikáció javára. A kiállításon szakmai regisztráció könnyíti meg a kapcsolatfelvételt a kiállító cégek és partnereik munkatársai között. Az ötnapos rendezvény első három napján a szakmai programok dominálnak, pénteken és szombaton azonban megnyílik a kapu az érdeklődő nagyközönség és a szervezetten érkező diákcsoportok előtt is.

„Hordozható nagyszámítógép"

Az Attachmate megjelent a PalmFrame nevű termékkel, amely alkalmas arra, hogy a PalmPilot és a Windows CE operációs rendszerű mobil eszközökről a vállalat központi számítógépén lévő adatokat elérjék, olvasni és módosítani tudják. A vállalat „hordozhatóvá tétele" által javulhat a munka hatékonysága, pontosabbá és frissebbé válhatnak az adatok, többen dolgozhatnak a központi irodáktól távol. Az utazó munkatársaknak például nem kell űrlapokat kitölteniük, adatokat utólag beírniuk, mert a központi számítógéppel közvetlenül összekapcsolódó PalmFrame gondoskodik az információ gyors továbbításáról, kiiktatva egy csomó papírmunkát, csökkentve a bérköltséget.

A PalmFrame rendszer magja a világhálón át történő elérést támogató Attachmate termék, az Attachmate e-Vantage Host-Publishing System. A központi adattárból kapott adatokat HTML formátumú adatállománnyá konvertálja, és azt az AvantGo Server továbbítja a központból a kéziszámítógép kijelzőjére. A kliensre telepített PalmFrame az üzleti munkafolyamatokhoz igazítható rendszer, alaptechnológiája ugyanaz, mint a kiszolgáló rendszeré, de a kéziszámítógépbe betöltött alkalmazás a feladat jellegének felel meg. Az adatok olyankor mozognak a kéziszámítógép és a gazdagép (nagyszámítógép, AS/400 vagy Unix szerver) között, amikor a mobil készülék a tartójában van, és előzőleg szinkronizációs módra lett állítva. A szinkronizáció azonban drót nélküli internet-kapcsolaton keresztül is megoldható.

A Texas Instruments tanít

A Texas Instruments (TI) a magyar értelmiség régi jó ismerőse, főleg a tudományos kalkulátorok révén. A cég 1998-ban átszervezte itteni értékesítési csatornáit, és az Euro-Profil lett a forgalmazás gazdája. A TI alapvető stratégiája azonban nem változott, elsődleges céljának az oktatási szféra kiszolgálását tekinti. Olyan intézményekkel épített ki szoros kapcsolatot, mint a Budapesti Tanítóképző Főiskola, az ELTE Tanítóképző Főiskolai Kara vagy az ELTE TTK Matematikai Módszertani Csoport. A tanárok képzésének és az új módszerek megismertetésének elősegítésére hazánkban is megalakították a T3 szervezetet (Teacher Teaching with Technology).

A tanítást és tanulást segítő eszközök közül a közelmúltban mutatták be a TI 15 modellt, amellyel az alsótagozatos oktatás



igényeit igyekeznek kielégíteni. A komplexebb feladatok megoldását a flash-memóriával ellátott gépek szolgálják. Ilyen a TI 30XII, a TI 92 Plus illetve a CBL 2. Az utóbbiakban mód van a programok cseréjére, így az adott feladathoz jobban lehet igazodni, de lehetőség van saját programok készítésére is. Az internetről számos program letölthető, köztük (utalva itt a hónap témájára) a lokalizálást megoldó csomagok. A letöltött, illetve a PC-n fejlesztett alkalmazások közvetlenül áttölthetők a TI eszközökbe. A fejlesztést saját SDK (Software Development Kit) segíti.

Mágneses szigetek

A kutatók kitartóan dolgoznak azon, hogy a növekvő adattárolási igényekkel lépést lehessen tartani. Az IBM-nél most olyan új eljárást fedeztek fel, amelynek felhasználásával a számítógépek merevlemezein és más háttértárolókon ugyanazon a felületen a korábbinál mintegy százszor több adatot lehet majd elhelyezni. A winchestereken az adatrögzítés úgy történik, hogy a lemez felszínének meghatározott pontjait a tárolandó információknak megfelelően különböző irányban mágnesezik, majd olvasáskor ezeket a mágnesezettségi irányokat tapogatják le. A tárolókapacitás tehát attól is függ, hogy milyen sűrűséggel tudjuk ezeket a mágneses szigeteket a korongon elhelyezni. Az új eljárásnak köszönhetően a mágneses szigetek (nanoszemcséknek is nevezik őket) feleakkorák lehetnek, mint korábban, emellett egyenletesebb a szemcsenagyság is, ami a pontosabb leolvasást teszi lehetővé. Ez utóbbi jelentősége nem lebecsülendő, eddig ugyanis 1000 mágneses szigeten lehetett egy bitnyi információt tárolni ezentúl pedig az IBM reményei szerint egy bitnyi információ rögzítéséhez egyetlen nanoszemcse (azaz egyetlen mágneses sziget) is elegendő lesz.

Yamaha CD-író 8x-os újraírással

A Yamaha az idei CeBIT-en bemutatta CRW 8824 típusú CD-író/újraíró berendezést, amely nyolcszoros CD-újraírási teljesítményre képes. A CD-író repertoárjába az összes fontosabb formátum beletartozik, köztük az egyre népszerűbb CD-Text is, amellyel a zenei számok mellett a dalok címei, szerzői és egyéb információi is elhelyezhetők, illetve lejátszáskor kijelezhetők. Elsőként az SCSI-3 (Ultra SCSI) csatlakozású belső egység lesz kapható, majd ennek külső változata, és végül az ATAPI/EIDE felületű CD-író. A 4 MB-os belső memória az adatelfogyásból származó pufferkiürülési hibát hivatott kiküszöbölni, a mintegy 140 milliszekundumos átlgos hozzáférési idő pedig a gyorsaságot biztosítja. A Yamaha cég hivatalos magyarországi nagykereskedője az egységet magyar nyelvű leírással, CD-író programmal és belső SCSI kábellel hozza forgalomba.

A leglátogatottabb portál

A magyar webportálok közül jelenleg a MatávNet által működtetett Origo a leglátogatottabb, hétköznapokon átlagosan 65 000, munkaszüneti napokon 45 000 nézővel. Az ingyenes levelezést biztosító Freemail több mint 300 000 postafiókos felhasználót szolgál ki, a SzoftverBázis az indulása óta eltelt fél évben fél millió szoftver letöltését regisztrálta. A MatávNet égisze alatt működik a legnagyobb hazai keresőrendszer, az AltaVizsla, továbbá a Fókusz Online könyváruház is.

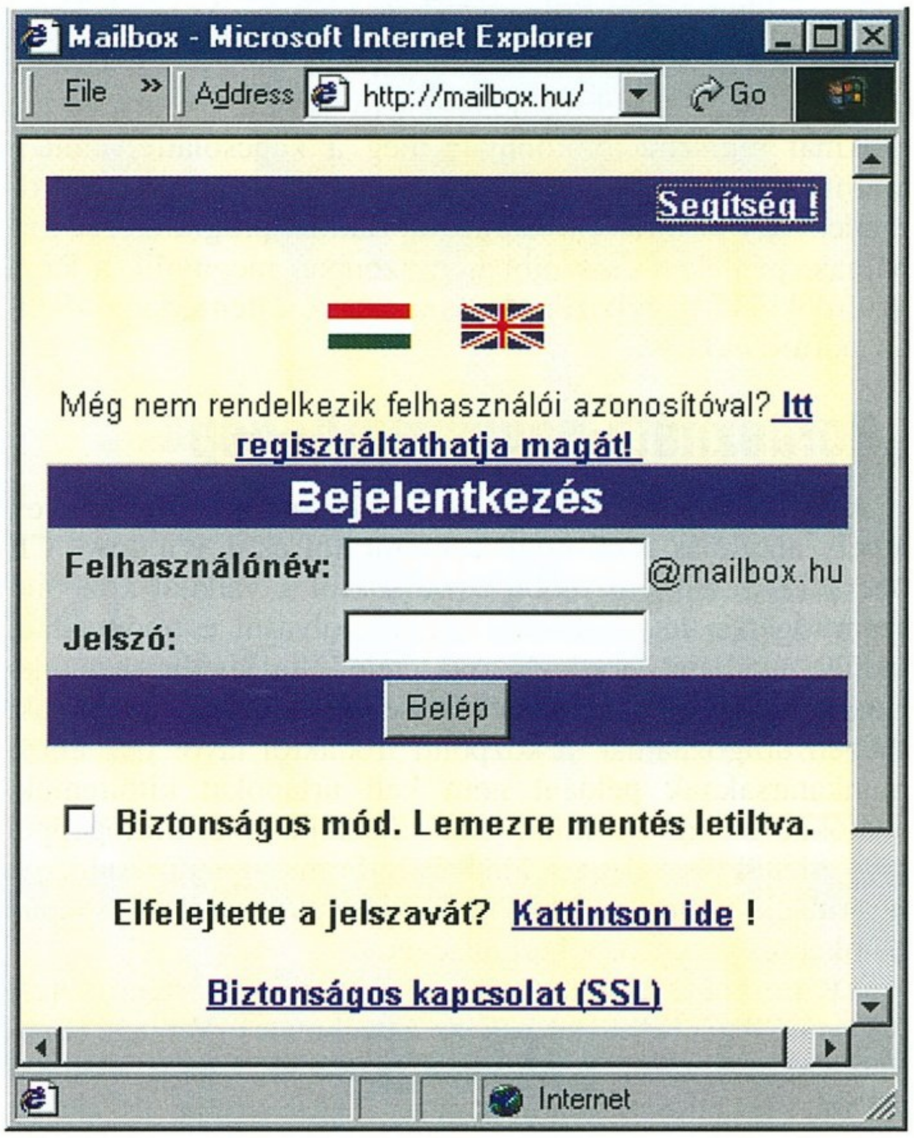
Az Origónál márciusban további új szolgáltatásokat vezettek be. A nemrég felújított Társalgó mellett elindult a Fórum (forum.origo.hu), amely feltehetően az Internetto és az Index hasonló vitarovatainak babérjaira pályázik. Ez is moderált rovat, azaz minden témát (topikot) az Origo valamelyik szerkesztője felügyeli, hogy a hozzászólások ne sértsenek jogszabályokat és etikai normákat.

Az Origo s.k. szintén új lehetőség, közvetlenül is elérhető az sk.origo.hu címen. Ennek segítségével minden látogató kialakíthatja testre szabott saját nyitóoldalát, beállítva, hogy az nála milyen szolgáltatásokkal jelentkezzen be. A kiindulási képernyőn elhelyezhetők például a friss hírek és a sporteredmények, a filmbemutatók, az aktuális szavazások, a saját linkek stb. Másrészt ami nem fontos, az el is tüntethető onnan. Az Origo s.k. oldalak tartalmának módosítása a dinamikus HTML (DHTML) technológia segítségével végezhető el, a szükséges elemeket a témák listájából csak oda kell húzni egérrel a kívánt oszlopba. A rendszer a régebbi böngészőprogramokkal internetezőknek is kínál megoldást a módosítások elvégzésére.



Új ingyenes levelezőrendszer

Minden eddiginél nagyobb tárhelyet biztosító új levelezőrendszert indított a Mail.Hungary. A felhasználónként 10 MB-nyi „levéltár" WWW és POP eléréssel egyaránt használható. A szolgáltatásban szerepel egy érdekes továbbítási (forward) megoldás is, amellyel a címzésen kívül a levélnek csak a tárgy mezőben lévő szövege jut el a megadott helyre. Így a mobiltelefonos felhasználók, akik szolgáltatójuknál e-mail címmel rendelkeznek, automatikus SMS üzenetet kaphatnak levelük érkezéséről és annak témájáról, maga a levél azonban nem kerül át a mobil címre és ott nem foglalja a kvótát. Egy ügyes szűrő beállításával az „SMS levelek" könnyedén törölhetők a mobil szolgáltatónál, békén hagyva a többi levelet. (További információk a http://mailbox.hu címen.)



Koffein a Telnetnél

A Telnet Magyarország Koffein nevű új internet-előfizetési csomagja a hazai piacon elsőként alkalmazkodik a Matáv által bevezetett átalánydíjas éjszakai és hétvégi hálózatelérést kínáló MindenkiNet csomaghoz. A Koffein lényege, hogy nettó 3600 forintos havi díjért munkanapokon este 6-tól reggel 7-ig, hétvégeken délután 3-tól reggel 7-ig korlátlan számú és időtartamú internet-kapcsolatot tesz lehetővé. A Matáv MindenkiNet csomagjával együtt a fenti időszakban megvalósított átalánydíjas internetezés havonta csak nettó 6800 forintba kerül. A Koffein csomag további sajátossága, hogy a kedvezményes időszakokon kívül is bármikor használható, nettó 360 Ft/óra díjért. További információk a http://www.telnet.hu honlapon.

Videómánia

Tömörítőkódolás és dekódolás

A világ tovább multimédiásodik. Egyre több helyen merül fel a digitális videók lejátszásának és készítésének igénye, ám aki a számítástechnika ezen területére merészkedik, meglehetősen zavaros viszonyok között találja magát. Az e havi közkinccsel a digitális videó szabványainak dzsungelében szeretnénk ösvényt mutatni.

Minden videófelvétel döntő technikai ismérve, hogy milyen kódolóval készítették. (Ezen nem valami titkosítást kell érteni, hanem a tömörítési módot.) A tömörítési kódolás határozza meg a fájl hosszát, minőségét, a lejátszáshoz — a dekódoláshoz — szükséges processzorteljesítményt stb. Ezért logikus lenne a videóállományokat a kódolás módja szerint kategorizálva úgy elnevezni, hogy a kiterjesztés utaljon a kódolási módszerre. Sajnos ez nem így történik, a sokféle kiterjesztés nem egyértelmű, ugyanaz az elnevezés különböző formátumokat takarhat, és ugyanaz a formátum meghúzódhat eltérő kiterjesztések mögött.

AVI (Audio Video Interleave)

Ez a „formátum" a Microsoft találmánya, és az elmúlt években de facto szabvánnyá vált a windowsos videóformátumok körében. Jellemzője, hogy nincs meghatározva a kódolás módja, a gyártók különböző saját AVI formátumokat definiálhatnak. Hogy egy AVI mégis milyen kódolású, azt a fájlban található, négy karakterből álló azonosító mondja meg. Ha a gépen telepítve van az adott kódrendszer kódolója, akkor lejátszható a videó, ellenkező esetben egy „Could not find vids:xxxx" üzenettel örvendeztet meg, ahol az xxxx a négybetűs kód. Hogy egy rendszerben mely kódolók vannak telepítve, azt 16 bites Windows esetén a system.ini, 32 bites Windows esetén a registry tartalmazza a drivers, illetve a drivers32 szekcióban levő bejegyzések között.

Tömörítetlen videó

Kódja: DIB (Microsoft Device Independent Bitmap) vagy RAW. Ezt minden rendszer kódoló nélkül le tudja játszani, könnyen szerkeszthető, viszont rendkívül nagy a helyigénye, ezért a legritkább esetben használják. Természetesen ennek a formátumnak is vannak alváltozatai, melyek elsősorban a felhasználható színek számában és kódolásában térnek el egymástól.

Microsoft Run Length Encoding

Kódja: MRLE. Tömörített, viszont ezzel a kódolással csak 256 színt tartalmazó videót lehet készíteni. Nem tömörít hatékonyan, de nem is kell neki nagy processzorteljesítmény, így szerkesztéshez is jól használható. Windows 3.1-hez a Video for Windows 1.1 készletet kell telepíteni, a 32 bites Windowsokban pedig már része a rendszernek.

Fájl: Vfw11e.exe

Microsoft Video 1

Kódja: MSVC, néha CRAM. Ezzel a kódolóval már 16 bites, tehát akár 65 ezer színt tartalmazó videót is lehet készíteni, a teljes (True Color) színpalettát azonban ez sem támogatja. A 16 bites változat szintén megtalálható a Video for Windows 1.1 készletben, a 32 bites változat pedig a 32 bites Windowsok alaptartozéka.

Fájl: Vfw11e.exe

Intel Real Time Video 2.1

Kódja: RT21. Van, aki Indeo 2.1-nek nevezi ezt. Mostoha a sorsa. 16 bites verziója a VFW 1.1e-ben megtalálható, de úgy tűnik, hogy a Microsoft már nem támogatja, mert 32 bites verziója ismereteink szerint nincs.

Fájl: Vfw11e.exe

Intel Indeo 3.1/3.2

Kódja: IV31 és IV32. Az előzőekhez hasonlóan a 16 bites változat a VFW 1.1e készlet része, míg a 32 bites megtalálható a 32 bites rendszerekben. Azonos kategóriába eső társaihoz képest jó minőségű, viszont nagy számítási teljesítményt igényel.

Fájl: Vfw11e.exe

Cinepak

Kódja: CVID. Korábban ez volt a leggyakrabban használt AVI kódoló. Algoritmusa ugyanazon alapul, mint az Indeo 3.x-é, és bár az jobb képminőséget ad ugyanakkora tömörítéssel, a Cinepak lejátszásához sokkal kisebb CPU teljesítmény is elegendő. Eredetileg a Macintosh gépekhez készítette a SuperMac Technology, de ma már három változata is létezik: az eredeti SuperMac 16 bites, valamint a Radius által írt 16 bites és 32 bites. Mindhárom ingyenesen hozzáférhető Windows és Macintosh platformokon (a Mac esetében mint QuickTime Video). Szintén része a VFW 1.1e-nek, a 32 bites Windowsok pedig alapbeállításban támogatják a Radius 32 bites kódolót. Egy



továbbfejlesztett 32 bites Cinepak kódoló is létezik Cinepak Pro néven, Mac-re és Windowsra egyaránt.

Fájlok: Vfw11e.exe
CinepakProDemo12.bin (Mac)
CinepakProDemo12.hqx (Mac)
CinepakProQTX.zip (Win32)

Motion JPEG

Kódja: MJPG. A legtöbb videódigitalizáló rendszer ebben a formátumban dolgozik. Ennek két oka van. Egyfelől minden képkockát JPEG-ként tömörít, ami viszonylag kis méretet eredményez, ugyanakkor ez a formátum még nem használja ki a képkockák hasonlóságait, hanem minden képkockát egyenként elment, így a videófolyam jól szerkeszthető. Ebből a formátumból is létezik három megoldás. A Microsoftnak van rögtön kettő: a Microsoft Motion JPEG Codec és a JPEG DIB Format, emellett van saját ajánlása az OpenDML Avi File Format Extensions nevű szervezetnek is. Több különböző szoftvermegoldás tartozik hozzájuk. Készített egyet a Paradigm Matrix, ez azonban csak OEM szállítók számára elérhető, a Morgan Multimedia cégé időkorlátos. A MainActor terméke a Main Concept, melynek van MMX támogatása, a Pegasus Imaging szoftvere pedig a PIC Video. Ezek mindegyike csak 32 bites platformon használható.

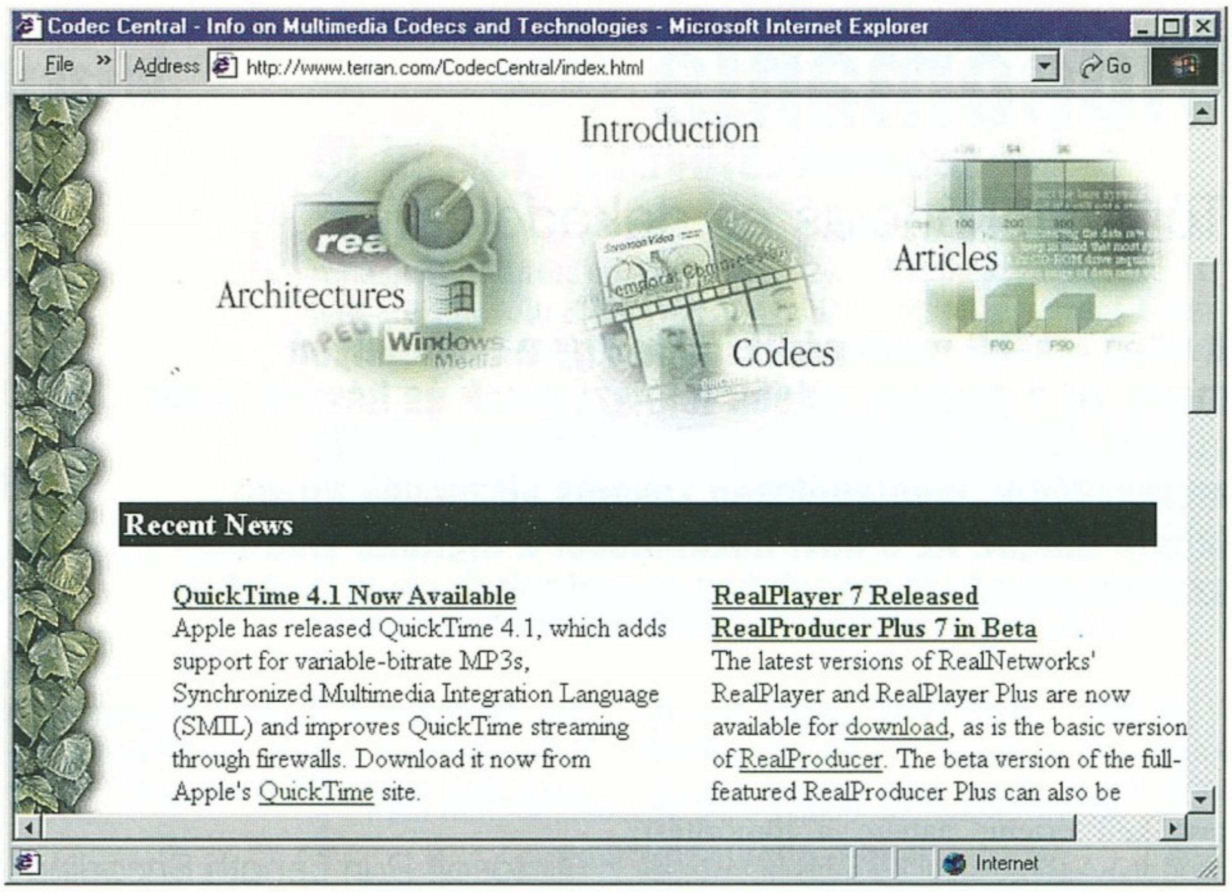
Fájlok: M3jpeg.exe (Morgan)
Picvideo2.exe (Pegasus)
Mcjpg28b (MainConcept)
Mcjpg30a (MainConcept Beta)

Editable MPEG

Kódja: XMPG vagy MPGI. Ebből a videóváltozatból is mindjárt kettő létezik, az egyiket a Xing Technologies (XMPG), a másikat a Sigma Designs (MPGI) készítette. A formátum jellemzője, hogy az ilyen videó MPEG tömörítésű ugyan, de minden képkocka egyben I-frame is, ezáltal nincs mozgáskompenzáció (az egymást követő képkockák hasonlóságainak figyelembevétele), nagyobb a méret, viszont képkockánként szerkeszthető. Tulajdonságai meglehetősen hasonlítanak a Motion Jpeg jellemzőire.

VDOWave vagy VDOLive

Kódja: VDOM, VDOW. Wavelet alapú kódoló, amelyet a Microsoft licencelt a Netshow nevű termékéhez. Két változata van, a VDOWave 2.0, amelyben az időegység alatt átvihető adatmennyiség (bit rate) kötött, és a VDOWave 3.0, amely már képes rugalmasan kezelni a továbbítandó adatmennyiséget. (Ez utóbbinak a kódja a VDOW.) Az összehasonlítások szerint a VDO



különösen a kis sávszélességet igénylő és gyengébb minőségű videók esetén feltűnően hasonlít az MPEG-1 kódolású videókra. Windows 95 alatt a Netshow 2.0 kliens vagy a Windows Media Player telepíti a lejátszáshoz szükséges dekódolót, míg a Netshow tools telepíti a VDOWave formátumú videók készítéséhez alkalmas eszközöket. Egészen pontosan a Netshow már integrálódott a Windows Media Technologies koncepcióba, így nincs külön Netshow client, helyette van a Windows Media Player.

Fájlok:
vdolive.macplayer.3.0.sea.hqx (Mac)
vdplayer16.exe (Win 3.1)
nstools.exe (Netshow tools 3.01)

Indeo Video Interactive (Indeo 4.1)

Kódja: IV41. Az Intel Indeo 3.x-hez képest új, hatékonyabb kódoló algoritmus. Kis sávszélességet igénylő és gyengébb minőségű videók esetén ez is nagyon hasonlít az MPEG-1 kódolású videókra (bár nem azt a kódolást használja).

Indeo Video Interactive 5.0

Kódja: IV50. Tovább tökéletesítették az algoritmust, jobb a képminősége (ugyanolyan tömörítettség mellett), viszont nagyobb teljesítményű processzort igényel. Ez a változat már támogatja a videók lejátszásának a letöltés befejezése előtti megkezdését — ennyiben a videófolyam formátumokhoz hasonlít (amilyen a RealVideo vagy az ASF). Jelenleg több változata is van, 5.0, 5.05 és 5.10 verziószámokkal. Az Intel egységes telepítőcsomagot kínál az Indeo összes változatával.

Fájlok:
Iv5setup.exe
(Win32 Indeo 2.x, 3.x, 4.x, 5.x AVI codec)
Indeo_Video_5.smi.bin
(Mac Indeo 5 támogatás QuickTime 4-hez)
Indeo_Video_5.smi.hqx
(Mac Indeo 5 támogatás QuickTime 4-hez)
IndeoVideo5.zip
(Win32 Indeo 5 támogatás QuickTime 4-hez)
iv5pdpub.exe
(Progressive Download Publisher Tool Demo)

MPEG-4

Kódja: MPG4, MP42, MP43. A jövőt tekintve talán ez a legjelentősebb az összes közül. Kiváló tömörítési arányt kínál szerényebb minőségi igények és korlátozottabb adatátviteli lehetőségek esetében éppúgy, mint a „felsőbb osztályban". Támogatja a videófolyamként való felhasználást is. Három változata létezik (version 1, version 2, version 3), melyek mindegyike megtalálható a Windows Media Tools csomagban.

Fájl: wmtools.exe (win32)

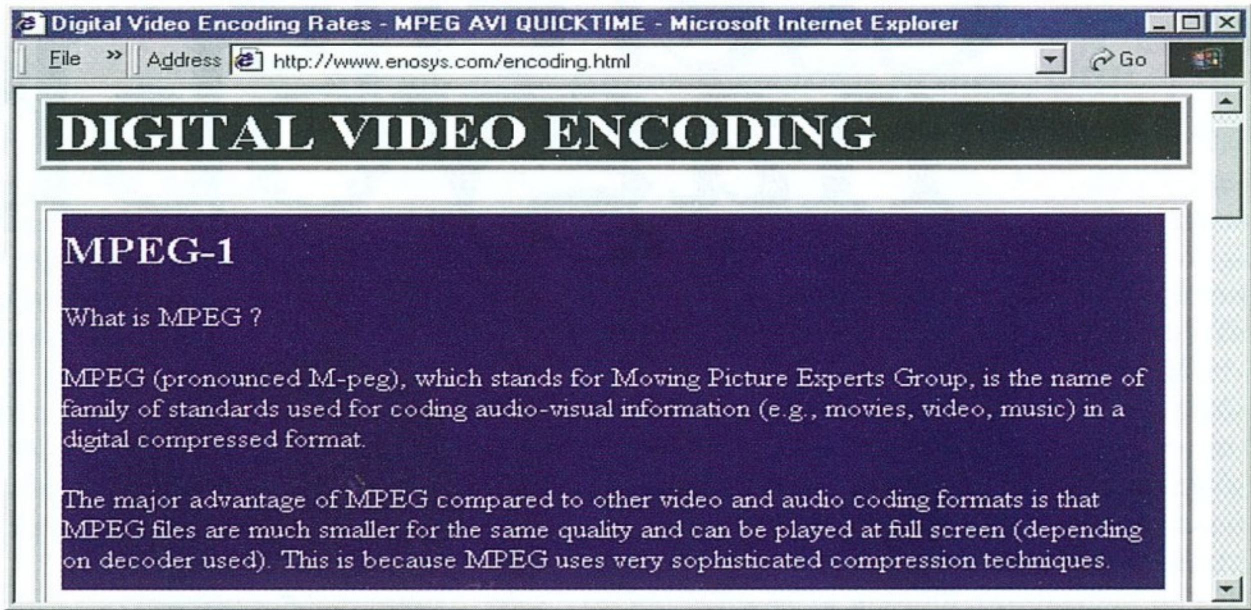
A fentiek voltak a legfontosabb AVI kódolók. Az AVI formátum egyik hátránya szerintem, hogy bárki készíthet hozzá kódolót, ezért nincs egységes csomagja, amely biztosíthatná, hogy az elkészült videó minden készüléken lejátszható legyen. Vannak speciális videódigitalizáló hardverekhez írt kódolók, speciális alkalmazásokhoz kialakított különleges AVI formátumok, és ezeket nagyrészt csak az adott hardverrel vagy alkalmazással lehet használni.

QuickTime

A QuickTime eredetileg az AVI megfelelője Macintoshra, de azért ez sem ilyen egyszerű. A QuickTime is támogatja például jónéhány AVI lejátszását, de van QuickTime Windows platformra is. Ráadásul a QuickTime nem egysze-

rűen videóformátum, hanem sokkal inkább multimédia-formátum. Támogatja a streaming felhasználást, a Pro változat rendelkezik szerkesztési funkciókkal, és nagyon jó minőségű.

A QuickTime használata kevesebb problémát okoz, mert egyrészt ez a formátum sokkal inkább centralizált (mindent az Apple tart kézben), másrészt a legújabb lejátszó mindig tartalmazza az összes szükséges kiegészítést (ez a Windows Media Playerről nem mondható el).



MPEG

Ez a VHS-sel megegyező minőségű általános digitális videószabványként jött létre — ez volt az MPEG-1. Elfogadható képminősége és nagyon jó tömörítési hatásfoka miatt a mai napig is nagyon sok helyen használják, pedig tetemes számítási teljesítményt igényel a dekódoláshoz. Önálló formátumként is létezik, az internetről letölthető videók jelentős része használja, és a videó CD-k formátuma is ez. Lejátszására alkalmas a Windows Media Player. Nagyon népszerű még a Xing technologies MPEG lejátszója.

Néhány éve jelent meg az MPEG-2 ajánlás, amellyel már fokozott minőségi igények is kielégíthetők. Ma leggyakrabban a DVD-k formátumaként lehet találkozni vele. Lejátszásához nagyon gyors processzor vagy speciális hardver szükséges.

Hivatalosan még nem jelent meg, de néhány helyen már használják a legújabb MPEG formátumot. Az MPEG-4 (hármas változat nem volt) megalkotásakor az internet igényeit tartották szem előtt. Tömörítési hatásfoka rendkívül jó, képes streaming formátumként viselkedni, kiválóan skálázható: kis méretű és egyszerű kivitelű videót éppúgy lehet ebben a formátumban készíteni, mint valamivel nagyobb helyigényű, de rendkívül jó minőségűt.

Néhány hete egyik kollegám megjegyezte, hogy minek ennyi formátum, amikor előbb-utóbb minden MPEG lesz. Ez a tendencia valóban megfigyelhető. Jellegében jónéhány AVI kódoló is az MPEG felé közelít, van Motion Jpeg és Editable Mpeg „AVI" formátum is, emellett a QuickTime használja az MPEG kódolót, és a DVD-ket is MPEG-2-vel kódolják. Ráadásul az MPEG önálló formátumként is létezik. A jövőben valószínűleg az MPEG-1 megmarad az alacsony (a VHS-ével azonos) minőségi követelményeket támasztó alkalmazásokhoz, az MPEG-2 a házimozi igényeit elégíti ki, míg a web sokszínűségének az MPEG-4 tud majd leginkább megfelelni.

Fájlok: xmp33at.exe (Xing player)
xme220t.exe (Xing encoder)

ASF

Sokak szerint az ASF nem igazán jó semmire. Bár a videókonferenciákat támogatja, nagy processzorteljesítményt igényel, és ami a legfőbb kritika ellene, nagyon rossz minőségű. Azt csak kevesen tudják, hogy az ASF videókódolója valójában MPEG, mégpedig MPEG-4. Ennek megfelelően kiváló képminőségű ASF-eket is lehet készíteni, más kérdés, hogy az ilyenek ritkák. Helyigénye azonos képminőség esetén csak mintegy fele egy MPEG-1-es videóénak. Az MPEG-4 jelenleg legjobban használható implementációja talán éppen az ASF, mert ehhez lehet a legszélesebb körben alkalmazható szerkesztő és konvertáló eszközöket találni, lévén azok a Windows Media Tools csomag részei.

Fájl: wmtools.exe (win32)

Nagy Tamás
bigtom@avalon.aut.bme.hu

Xircom: RealPort2 integrált PC-kártya

A Xircom Europe NV-nek, a mobil adatelérési megoldások vezető szállítójának PC-kártyája a RealPort integrált PC-kártyacsalád új tagja, amely a noteszgépek felhasználóinak készült, és a kiegészítő kábelek nélküli adatelérési lehetőségek még szélesebb skáláját nyújtja. A Xircom szabadalmazott, integrált csatlakozórendszerével ellátott RealPort2 kártyát önmagában, illetve más RealPort2 kártyákkal kombinálva is lehet használni. A kártya konstrukciójának köszönhetően a noteszgépek felhasználói az elérési technológiák széles választékából (66 kbit/sec-os modem, Ethernet, ISDN, DSL, GSM, PDC, telefonvezetékes HomePNA hálózatok vagy Bluetooth) mindig a nekik leginkább megfelelőt vehetik igénybe.

A felhasználó egy sor külön funkciókártyából választhatja ki a neki szükségeseket, és azokból teljes mértékben saját igényeihez szabott, kiegészítő kábelek nélküli megoldást állíthat össze. Elég két kártyát egymásra csúsztatni, és máris egyetlen eszközként lehet a két RealPort2 adaptert használni, amely bármely szabványos Type III PCMCIA bővítőhelyhez csatlakoztatható. A főbb jellemzők: szabványos RJ-45-ös Ethernet és RJ-11-es telefonkábeleket fogadó integrált csatlakozók; egyedül vagy másik RealPort2 adapterrel kombinálva egyaránt használható PC-kártyák; nagy teljesítményű 56 K-s modem, 10/100 Mbit/sec-os Ethernet LAN, ISDN, GSM és a PDC, valamint olyan jövőbeli technológiák, mint a vezeték nélküli Ethernet, DSL, telefonvezetékes HomePNA hálózatok és a Bluetooth; GlobalACCESS funkciók (köztük a CountrySelect szoftver) a megbízható, egyszerű kapcsolatteremtés céljára; energiatakarékos funkciók a noteszgépek akkumulátoros üzemidejének meghosszabbítására.

RAD: WebRanger router

Ha egy iroda vagy telephely lokális hálózatát, telefonközpontját szeretnénk másik telehelyhez vagy internet-szolgáltatóhoz csatlakoztatni, megoldást nyújtanak rá az izraeli RAD cég WebRanger termékei. A WebRanger kompakt internet/intranet-hozzáférést biztosító router, amely kis- és közepes méretű irodák, cégek kapcsolódását teszi lehetővé hagyományos aszinkron modemes, szinkron bérelt vonali, ISDN vagy Frame Relay hálózaton keresztül. Az internet-hozzáférés mellett kisebb magánhálózatok összekötése is megvalósítható, akár IPX protokollon keresztül. Az internet-szolgáltató útválasztó berendezéseivel szabványos Frame Relay, SLIP, CSLIP vagy PPP, illetve Multilink PPP kommunikációs protokollt használhat, beleértve az utóbbi esetben szokásos PAP/CHAP autentikációt is. A WAN oldali fizikai interfész többek között lehet aszinkron V.24 — 2,4 kbps-től 15 kbps-ig —, szinkron V.24 és ISDN BRI „S" interfész esetén 128 kbps-ig, szinkron X.21, V.35, V.36 és RS-530, illetve G.703 E1-es kapcsolat esetén 2 Mbps-ig. Korlátozott hozzáférést igénylő felhasználók esetében a kapcsolt vonalak felépítése és bontása a forgalomhoz időzíthető és méretezhető. Az egy vagy két, 10 Mbps-os Ethernet LAN interfészek csavart érpáras UTP, koax, BNC vagy AUI felületek lehetnek (igény esetén E1-es alcsatornával is rendelkezhet privát hálózatokban alközpontok összekapcsolására).

A biztonságosabb kapcsolódást a világhálózatra tűzfal funkciókkal is támogatja a RAD új eszköze, amely emellett biztosítja a nem legális címtartományú hálózatok internet-hozzáférését is egyetlen legális IP címmel. Az üzembe helyezést és felügyeletet segíti a hagyományos konzolportos konfiguráció mellett a Telnet és a HTTP böngészőkkel megvalósítható paraméterezés is. A központosított hálózatfelügyelet esetén használt SNMP protokoll támogatása szintén megoldott. További információ a hazai forgalmazó Lanex Kft-től kapható; tesztelési lehetőséget is biztosítanak.

Novell: IP-kezeléssel bővített ManageWise

A Novell bejelentette, hogy kapható a ManageWise 2.7 hálózatfelügyeleti program legújabb verziója. A ManageWise-zal a szervezetek egyetlen pontból figyelhetik többplatformos hálózataikat. A 2.7-es verzió natívan kezeli a „tiszta IP" szegmenseket és IP hálózatokat is, nem szükséges a továbbiakban a NetWare IPX-kompatibilitási módja, így leegyszerűsödik és felgyorsul a szerverfelügyeleti feladatok ellátása. A rendszergazdák hatékonyabban felügyelhetik gyorsan növekvő IP-hálózataikat, ugyanakkor továbbra is kézben tarthatják a többprotokollos környezeteket, valamint a vegyes (NT–NetWare) hálózatokat. A ManageWise 2.7 további újdonsága, hogy egyazon termékben biztosítja a NetWare és NT szerverek kezeléséhez szükséges összes funkcionalitást (a korábbi változatban az NT felügyeletét külön kellett megvásárolni). Megtalálható a dobozban a McAfee VirusScan és NetShield programcsomagja is, mégpedig hat hónapos frissítési előfizetéssel.

SMC: gyors hálózati kártya

A kis- és közepes méretű LAN hálózatokhoz készített termékekkel foglalkozó SMC Networks hálózati eszközgyártó cég egyik új terméke a TigerCard 1000 SMC 9462-SX. Ez nagy megbízhatóságot követelő hálózati környezetben is alkalmazható, az 1000SX-SC csatlakozás újszerűen köti össze a szervert és a felső kategóriájú munkaállomást, amelynek eredményeként jelentősen nő a hálózati teljesítmény. Az új hálózati eszköz legfontosabb jellemzői: 32/64 bites PCI Bus Master kártya, optikai SC csatlakozó 2 Gbps sebességű átvitelhez full-duplex üzemmódban, 128 VLAN támogatása, diagnosztikai LED kijelzők. Az SMC új kártyája illeszkedik a Windows 95/98, Windows NT 4.0, Windows 2000, Novell NetWare 3.12, 4.11, 5.0 és Linux operációs rendszerekhez. A meghajtóprogram a cég www.smc-europe.com weblapjáról is letölthető, ahol emellett termékleírások, prezentációk és termékfotók is találhatók.

3Com: e-Networks újdonságok

A 3Com User Group megalakulása alkalmából, február 18-án rendezett budapesti konferencián a cég több újdonságát, köztük az e-Networks termékeket is bemutatták. A 3C 905C-TX-M jelű, EtherLink 10/100 kártya munkaállomások ellenőrzésére szolgál, új teljesítménymonitorozó algoritmussal. A hálózatot az a fontos alkalmazások szempontjából optimalizálja, és prioritási sorrendet állít fel a forgalomra az összeütközések csökkentése érdekében. A 3C 980C EtherLink Server 10/100 kártya gyors szerverhozzáférést biztosít, kezeli bármelyik gyártó szabványalapú kapcsolóját, alkalmazásával hibatűrő kapcsolat építhető ki a szerver és a hálózat között. Szerverenként maximálisan 64 VLAN-t képes támogatni. A 3Com SuperStack II Switch 3300 XM típusú kapcsolója maximum 24 Fast Ethernet 10Base-T/100Base-TX porttal rendelkezik. A berendezésben külön mátrixport látja el a 3300-as és 1100-as jelű kapcsolók közötti rétegelési funkciót. Az eszközbe multimédia funkciókat is beépítettek (GMP snooping, 802.1 P/Q). A 100Base-FX porttal rendelkező plug-in bővítőmodulok teljes vagy félduplex üzemmódban működhetnek. Szintén 10/100Base-TX típusú kapcsoló a 3Com SuperStack II Switch 3900 típusú kapcsolója, amely kétféle, 24 vagy 36 portos kivitelben kapható, és amelyet Gigabit Ethernet csatolófelülettel is elláttak. A Cat 5-ös kábelezést támogatja a SuperStack II Switch 9100 típusú kapcsoló, amely 6 darab 1000Base-T és 2 db 100Base-SX porttal rendelkezik. További fontosabb jellemzői: 128 K-s MAC címezhetőség, porttrönkölés, csomagszűrés, snooping funkció, 8002.1 P/Q alapú VLAN-ok támogatása, porttükrözés, 4 db RMON csoport, QoS megoldás.

Kovács Attila

Novell szerverek és a Win2k

Az együttműködés új útjai heterogén hálózatokban

Egy hálózatos operációs rendszer új verziójának megjelenésekor érdemes szemügyre venni, miképpen használható az vegyes hálózatokban. Különösen gyakran felmerülő kérdés ez a Novell NetWare és a Microsoft Windows esetében. Ezek a rendszerek a szerverek feletti uralom megszerzésében konkurenciát jelentenek egymásnak, de utána a kialakult helyzetben többé-kevésbé békés egymás mellett élés alakulhat ki köztük. Ennek egyik magyarázata, hogy a Novell a hálózat használatát, adminisztrációját szolgáló számos olyan programot fejlesztett ki, amely fel van készítve, hogy a munkaállomásokon Windows operációs rendszert kezeljen, hiszen sok esetben NetWare szervereket használ a windowsos munkaállomásokhoz, tekintve, hogy a Novell a desktop kategóriára ma már (sajnos) nem fejleszt operációs rendszert.

A Windows 2000 bevezetésekor felmerülő egyik gyakorlati kérdés, hogy a korábban kiépített NetWare rendszerek hogyan „tűrik el" a munkaállomásokon az Win2k telepítését. Annak ellenére ugyanis, hogy Microsoft (márcsak üzleti okokból is) elsősorban homogén hálózatra ajánlja a Windows 2000-et, nincs akadálya a szólógépre kliensként történő telepítésnek. Hasonló a helyzet, mint a Windows NT 4.0 esetében, amely üzembiztosabb volt, mint a Windows 9x, ezért gyakran telepítették annak a helyére.

Vegyesen

A döntő kérdés az, hogy a munkaállomásról a NetWare szerverek probléma nélkül használhatók-e. A Windows esetében a Novell rendszeresen ellátja a felhasználókat ilyen célt szolgáló kliensprogramokkal, általában szabadon letölthető formában. Ez nemritkán biztonságosabb kliens-szerver kapcsolatot teremt, mintha beépített kliensprogramok lennének.

A Novell a Windows 2000-hez optimalizált kliensprogramot csak a Win2k piaci megjelenését követő időszakra ígérte, ennek ellenére honlapján (http://www.novell.com) már 2000. januárban elérhető volt egy csomag, amely két programot tartalmaz. Az egyik a Novell Client for Windows NT/2000 v4.7, amely a dokumentáció szerint egyaránt használható a Windows 2000-rel telepített gépeken, és a korábbi Windows NT 4.0 verzión, ha az legalábbb SP 3-as frissítésű (CD-mellékletünkön megtalálható).

A másik a ZENworks kezdőcsomagja (ZENworks Starter Pack), amelyben már szintén megvan az új Windows verzió támogatása. (Részletesebb információk a http://www.zenworks.com címen olvashatók.)

Licencmegállapodások

A ZENworks egyike azoknak a programoknak, amelyek a munkaállomáson Windowst igényelnek. Amikor frissítik az operációs rendszert, és a program nem kompatibilis az új verzióval, a felhasználót és a rendszeradminisztrátort is érheti néhány kellemetlen meglepetés. Tehát a heterogén rendszerek szervereinek piacán a Novell kénytelen figyelemmel kísérni a Windows verzióváltásait, ha meg kívánja őrizni felhasználóinak bizalmát.

A Novell 2000 elején abban is továbblépett, hogy a NetWare 5.1-ben megjelent az NDS új verziója. Az NDS-t használó, a Windowst is kezelő, vagy azon futó programok esetében időszerű e kettős verzióváltáshoz való alkalmazkodás. (Az NDS továbbfejlesztésének eredményeiről, az eDirectoryról lapunk 2000. februári számában adtunk áttekintést.) A NetWare fejlesztése, és valószínűleg a bejelentés időzítése is jelzi, hogy a Novell természetesen a szerver oldalán sem kívánja figyelmen kívül hagyni az új Windows kihívását. (Ezzel már az 1999 márciusában megtartott Brainshare konferencián is külön foglalkoztak.)

Érdekes licencmegoldások is születtek a Windows 2000-rel kapcsolatban. A heterogén hálózatok támogatása érdekében a Novell 100 felhasználós NDS eDirectory licencet ad azoknak, akik a 2000. március 13-án bejelentett megjelenésétől számított 90 napon belül Windows 2000-res szervert vásárolnak. Hasonló ajánlatot kaptak azok is, akik 2000 januárjában a Sun Solaris 7-es verzióját vásárolták meg. Ha a felhasználók élnek ezekkel a lehetőségekkel, a nagyobb vállalati rendszerekhez rugalmas megoldásokat tudnak választani. Ezt a kínálatot szélesíti majd a Windows 2000-re szánt eDirectory verzió is, amelynek megjelenése 2000 első félévében várható.

Címtárverseny

Kényes kérdés, hogy a címtárak mennyire fogják támogatni egymás adatainak elérését. A Windows 2000-ben debütál a Microsoft Active Directory (AD) címtárszolgáltatás, amely belső logikájában (természetesen?) eltér az NDS alapú megoldástól. (Az ADSI rajta van mostani CD-nken.)

A két rendszerrel kapcsolatos összehasonlítás nem ennek a cikknek a tárgya, de az elkövetkező időszakban erre vonatkozóan is várhatók részletesebb információk. Addig is az érdeklődőknek ajánljuk a Novell szemszögéből a http://www.novell.com/corp/strategy/fsd/matrix.html weboldalt, a Microsoft szemüvegén át pedig az AD technikai részleteiről a http://www.microsoft.com/windows/server/Technical/default.asp weboldalt.

Az kétségtelen, hogy mindkét rendszer támogatja az LDAP ipari szabványt, és az a közvetlen kapcsolódás teljes kiépítéséig áthidaló szerepet tölthet be a két címtár használatában. Azok pedig, akik arra is kíváncsiak, hogy mi a Novell véleménye a W2k-ról, nézzék meg a http://www.novell.com/advantage/w2k.html dokumentumot.

Simay Endre István

Higgy, és üdvözülsz!

A házhozszállított ingyenes reklámújságok egyikében találtam egy szórólapot, amelyen X Oktatási Központ hirdetett számítástechnikai oktatást. Szigorú követelményt támasztott: *„Tanfolyamainkra a jelentkezés feltétele az általános iskolai végzettség”.*

Akik erre a tanfolyamra rászánnak egyhavi átlagfizetésnek megfelelő összeget, azok a szórólap ígérete szerint megtanulják egyebek közt az *adatvédelmet...* Hogy mit? Vajon azt az ismeretenyagot, amelynek új fejleményeiről való tájékozódásunkhoz is tucatnyi weblapot kell áttanulmányozni? (Angolul, ami viszont nem tanfolyami követelmény.)

A *„lemez és állomány-műveletek”* kifejezés is nagyon jól hangzik, de mit takar? Talán a „drag and drop” technikát, amellyel a Windows 95 első elindítása után percek alatt önállóan is tisztában lehetünk?

A tanfolyam hallgatói *„továbbá elsajátítják a WINDOWS 95 használatát”* és persze *„megtanulják az OFFICE 97 WORD szövegszerkesztő használatát”.* Olyannyira, hogy a tanfolyam végén a hallgatók készek lesznek az E.V.R. vizsga letételére. (Nem tudom mit jelent, de nagyon jól hangzik.)

Tegyük fel, hogy a hallgató rájön, mennyire keveset tanult (és tud), ezért újabb egy havi fizetésnek megfelelő összegért beiratkozik a középfokú tanfolyamra is. Ez már a Microsoft Office csomag többi elemével foglalkozik. Ha csak az Excelre gondolunk, általános iskolai végzettséggel meg kell tudni csinálni benne mondjuk az SZJA-elszámolást. (Az APEH honlapjáról letölthető az ehhez Excelben megírt program.) De mi van a bonyolultabb feladatokkal, például egy kisvállalkozásnál szükséges adatfeldolgozással?

A hallgatókat talán megfogják azzal, hogy *„ez a bizonyítvány Európa-szerte egységesen elfogadott”.* És a tudás? Meg az általános iskolai végzettség? Meg a nyelvismeret?

A számítástechnikai oktatás ma olyan vadászterület, ahol hiányzik a korrekt tájékoztatás. Talán elhamarkodott állítás lenne, ha kijelenteném, hogy humbug a példaként említett és a többi hasonló tanfolyam. De azok, akik gyakrabban találkoznak e tanfolyamok „végtermékeivel”, talán megerősíthetnék — vagy megcáfolhatnák — következtetésemet. Írják meg, mi a véleményük.

Szondi Egon János
szondi@reak.bme.hu

Kérdések, kérdések...

Az utóbbi időben foglalkoztat egy jelenség, amelyre nem találok magyarázatot. Egyik ismerősöm gépe (Celeron 433 MHz, 128 MB RAM, Quantum Ultra-ATA66 merevlemez) állandóan lefagyott, és időnként olyan végzetes hiba keletkezett, hogy az egész Windows 98 operációs rendszert újra kellett telepítenie. Hosszas keresgélés után a Matrox Millennium G400 16 MB-os kártyát tettem meg bűnbaknak. A Vezérlőpult – Rendszer – Teljesítmény – Grafika ablakban kiiktattam a hardveres gyorsítást. Ezzel „megoldottam” a problémát, de megválaszolatlan kérdések egész sorával találtam magam szemben:

— Mi a hardveres gyorsítás szerepe a PC-n, Win98-ban?

— Mit jelent a fenti Grafika ablakban lévő csúszka négyféle beállítása a nincstől a teljesig? Miért jelent ez a csúszka orvosságot a váratlan és kiismerhetetlen lefagyások problémájára?

— Mi a szerepe a DirectX meghajtónak a grafikus gyorsításban?

— Mi a különbség a DirectX és az OpenGL között?

— Miért vegyen valaki méregdrága grafikus kártyát, ha utána szoftveresen le kell tiltania? (A ma megvásárolt kártyát másnap 70%-on visszaveszi az üzlet.)

— A Windowsra megírt gyári meghajtó optimalizálva van magára a G400 chipkészletre, míg a Corel Linux meghajtója egy általánosított meghajtó SVGA-ra. Azt jelenti-e ez, hogy a Linux nem használja ki a Matrox Millennium G400 kártya összes lehetőségét, és a rendszer azért is stabil ebből a szempontból, mert a hardveres gyorsítás a Linuxban ki van iktatva? Egyáltalán mennyit használ ki a Linux az ilyen nagy teljesítményű kártyák adottságaiból?

— Okozhatja-e az állandó, és nem kiismerhető lefagyásokat hardveres probléma, például memóriahiba magában a 128 MB RAM-ban vagy a Matrox kártya saját memóriájában?

— Hogyan tesztelhető a grafikus és a normál memória?

— Túlhajthatja-e egy grafikus alkalmazás a processzort, ami ennek következtében túlmelegszik és lefagy?

— Van-e a grafikus kártyáknak processzorigényük és fordítva? Azaz a fenti Matrox kártya esetleg nem elégszik meg a Celeron processzorral, hanem Klamathot szeretne?

— Milyen viszonya lehet az alaplapnak (ami jelen esetben Abit BE6-os) a grafikus kártyához, és fordítva?

— Mire jó az AGP slot, és miben különbözik a többitől?

— Milyen más szoftveres vagy hardveres tényezők idézhetnek elő lefagyásokat?

Mindezekről szívesen olvasnék az Új Alaplap hasábjain olyan szakértők tollából, akik választ tudnak adni.

Szaló István
ratiosoft@freemail.c3.hu

A Java jó ötlet

Szeretnék gratulálni a kiváló témaválasztáshoz, a Java sorozathoz. Jó ötlet volt elindítani, meg is vásároltam a januári és februári számot, és a többit természetesen szintén beszerzem. Örömmel látnám még a CD-n a Java tutorialt. Megnéztem az interenten, de tömörítve 7 MB feletti méretű, ezért a letöltéséről lemondtam. Más dokumentumoknak is nagyon örülnék, még ha angol nyelvűek is. Java programok forráskódjának közlése ugyancsak hasznos lehet kiegészítésként.

Nagyon szimpatikus, hogy foglalkoznak az alternatív operációs rendszerekkel. Ezek közül én a Linuxot használom, mindig izgatottan várom a CD-n az új programokat. Ehhez is van egy javaslatom: ha a wxWindows,wxGtk,wxpython felkerülhetne a lemezre, bizonyára érdekelne többeket is. Ez egy platformfüggetlen grafikus könyvtár, amely Win32 és Linux+Gtk alatt működik, C++ és Python nyelven pedig fejleszteni lehet hozzá.

Eddig a munkahelyemen olvastam a lapot, de a Java sorozat — és az egyre több érdekes cikk — arra ösztönzött, hogy saját példányot is vegyek. Az Új Alaplap egészen más, mint a többi lap a hazai piacon, tartalmi és formai tekintetben egyaránt. Nem a külföldi lapok hazai kiadása vagy utánzata, és szerintem így jó, ahogy van. Egy igazi szakmai lap. Sok sikert kívánok és további eredményes munkát!

Székely Krisztián
krisztian.szekely@hu.origin-it.com

Örülünk, hogy a Java sorozat kedvező fogadtatásra talált. A tanfolyamhoz készített illusztrációk mellett igyekszünk a CD-n egyéb forráskódokat és dokumentumokat is elhelyezni. A tutorialt most mellékeljük, az említett linuxos libraryt pedig legközelebbi korongunkon találhatja majd meg.

A szerk.

Magyar virtus

Új feltörnivaló mindig akad

„Lopni vagy nem lopni, az itt a kérdés..." az eredeti shakespeare-i idézet elferdítésével kitűnően jellemezhetjük a magyarországi helyzetet. Míg kívülről nézve az ország egyre feljebb kerül a „becsületességi" ranglistán, addig itthonról nézve a kép bizony a szürke számos árnyalatát mutatja.

Hogyan zajlik ma Magyarországon a programok beszerzése? Három fő típus különböztethető meg:

1. A legegyszerűbb magyaros mód az interneten található cracker és hacker honlapokról a (feltört) kereskedelmi szoftverek ingyenes letöltése és használata. Ha a program nem fut, sebaj, felhívják a gyártó hivatalos magyarországi képviselőjét, hogy segítsen, természetesen ingyen. Ha az szóba sem akar állni velük, akkor kijelentik, hogy ilyen szakmailag gyenge cégnek már régen tönkre kellett volna mennie, és különben is, milyen gyártó az, amely nem ad támogatást termékeihez. Hogy nem fizettek érte...? Ugyan már... szoftverért?

2. A második kategória inkább végigjárja a jól ismert helyeket, ahol néhány ezer forintért programokkal telepakolt CD-ket zsákmányolhat. Ha azok közt sincs ott a hőn áhított program, akkor meglátogatja a csempész cégek egyikét-másikát, hogy a megvásárolt hardver dobozában ugyan felejtsenek már benne egy floppyt vagy CD-t. Végső esetben pedig becserkészik barátaikat, akiktől egy legközelebbi hasonló szívesség fejében megkapják a kívánt verziót.

3. Az első két csoport meg sem érti a harmadik csoport viselkedését, azokét, akik jogtiszta szoftvert vásárolnak. Ők tudják, hogy a technikai támogatás, a betanítás, a rendszeres frissítés pénzbe kerül, és a termékkel együtt ezt vásárolják meg. Ez a kör egyre szélesedik, és az ő vásárlásaik és pénzük révén látják kis hazánkat kívülről egyre becsületesebbnek.

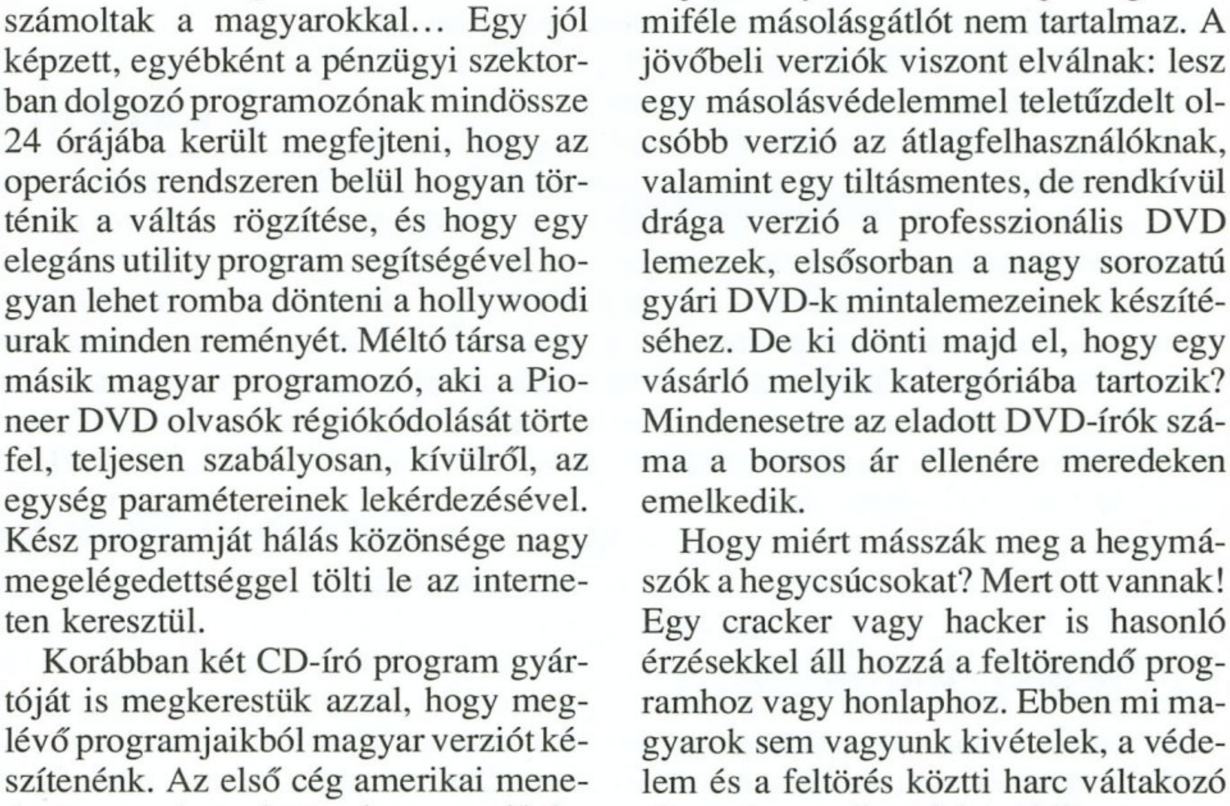
De nemcsak passzív szemlélői és haszonélvezői vagyunk a crackerek és hackerek népes táborának. Jellemző példa a DVD-ROM és a hozzájuk tartozó MPEG2 dekóderkártyák esete. Ismeretes, hogy a DVD fórum — a filmgyártók és forgalmazók nyomására — kitalálta a régiókódolást. A világot hat régióra osztották, és például Amerikában csak az 1-es, Európában és Japánban csak 2-es régiókódú DVD lemezeket szabad használni. A lejátszókba és a dekóderkártyákba pedig 5 alkalommal megváltoztatható régiófigyelést építtettek be. Az ötödik alkalom után a rendszer leragadt az legutolsóként beállított régiókódnál. Persze nem számoltak a magyarokkal... Egy jól képzett, egyébként a pénzügyi szektorban dolgozó programozónak mindössze 24 órájába került megfejteni, hogy az operációs rendszeren belül hogyan történik a váltás rögzítése, és hogy egy elegáns utility program segítségével hogyan lehet romba dönteni a hollywoodi urak minden reményét. Méltó társa egy másik magyar programozó, aki a Pioneer DVD olvasók régiókódolását törte fel, teljesen szabályosan, kívülről, az egység paramétereinek lekérdezésével. Kész programját hálás közönsége nagy megelégedettséggel tölti le az interneten keresztül.

Korábban két CD-író program gyártóját is megkerestük azzal, hogy meglévő programjaikból magyar verziót készítenénk. Az első cég amerikai menedzsere azzal utasította vissza az ajánlatot, hogy olyan megbízhatatlan piacon, mint a magyar, a honosítás csak növelné az illegális másolatok számát. A másik cég, a német Ahead — amely a professzionális felhasználók szerint a ma kapható legjobb programot, a Nerót készítette — másképp látta a kérdést. Ők hosszú távon gondolkodnak, és úgy vélik, hogy egyrészt a magyar verzió a nyelvet nem beszélő körnek is lehetővé teszi a CD-írást, másrészt a néhány havonta megjelenő frissítés, a program tudásásak teljes kiaknázása, és a közvetlen szakmai támogatás előbb-utóbb rávezeti a most még illegális felhasználókat a vásárlás előnyeire. Így a vásárlók birtokba vehetik a Storage System segítségével elkészült magyar nyelvű Nero CD-író programot.

Az illegális másolást és használatot nehezítő és segítő tábor sehol nem egységes. A rablóból gyakran lesz kitűnő pandúr, és világcégek sokszor alkalmaznak komoly fizetésért volt crackereket. Olvashattunk az USA állami szerveinél alkalmazásban álló megtértekről, akik a többszörösen védett szerverek biztonságát tesztelik. A másik oldal sem mindig tudja, hogy hol áll. Tipikus példa a Pioneer cég új DV.D-R írója, amelynek mai verziója még semmiféle másolásgátlót nem tartalmaz. A jövőbeli verziók viszont elválnak: lesz egy másolásvédelemmel teletűzdelt olcsóbb verzió az átlagfelhasználóknak, valamint egy tiltásmentes, de rendkívül drága verzió a professzionális DVD lemezek, elsősorban a nagy sorozatú gyári DVD-k mintalemezeinek készítéséhez. De ki dönti majd el, hogy egy vásárló melyik katergóriába tartozik? Mindenesetre az eladott DVD-írók száma a borsos ár ellenére meredeken emelkedik.

Hogy miért másszák meg a hegymászók a hegycsúcsokat? Mert ott vannak! Egy cracker vagy hacker is hasonló érzésekkel áll hozzá a feltörendő programhoz vagy honlaphoz. Ebben mi magyarok sem vagyunk kivételek, a védelem és a feltörés köztti harc váltakozó sikerrel, csendben folytatódik.

Ménesi Balázs
storage@mail.matav.hu



Netszabadság

A Nagy Testvér figyel(tetne) Téged (is)!

Paranoia: Az ebben szenvedő beteg teljesen logikusan gondolkodik és cselekszik, csak meg van róla győződve: valakik, valmiért el akarják pusztítani, üldözik. Ezt a meggyőződését többnyire alá is tudja támasztani a körülötte zajló események sajátos értelmezésével és csoportosításával. Kiegészítés: Igazán peches paranoiás az, akit végül el is kapnak...

Azért a bevezetőben e pár soros értelmező szótár, mert nem igazán tudom, hogy jól fejlett üldözési mániám, vagy az események menete juttatott el egy furcsa gyanúig az internet biztonságával kapcsolatban.

Amerikában az utóbbi időben sorozatosan támadták meg a legnagyobb tartalomszolgáltatók és keresőrendszerek szervereit. Pár órára, esetleg egy-két napra lebénították a gépeket, lehetetlenné tették azok normál üzemelését.

Legyinthetnénk is a dologra, mert ilyen támadások naponta érnek kisebb-nagyobb forgalmú internetes szervereket. Sőt, nyugodtan ki merem jelenteni: minden olyan gép, amely az internetre kapcsolódik, ki van téve hasonló támadásoknak, legfeljebb az elkövető kétbalkezessége (vagy mesteri tudása) miatt fel sem tűnik. Ha valaki veszi magának a fáradságot, és telepít olyan segédprogramokat, amelyek „megfogják" és naplózzák a Netről jövő támadásokat, betörési kísérleteket, az előbb-utóbb talál a naplófájlban ilyesmire utaló jeleket. Ennek lehetősége nyilvánvaló: az internetes kapcsolatokra használatos szoftverek túlnyomó többsége a trehányság és a kapkodva piacra dobás miatt hibákkal, biztonsági lyukakkal terhelt.

Sajátos módon az amerikai és néhány európai kormány a nagyszabású támadássorozat után egy-két napon belül előállt egy előkészített törvényjavaslattal „az internet megóvása érdekében". Közben a tömegmédiákban megjelenő cikkek, jelentések, riportok is azt igyekeznek sulykolni a fejekbe, hogy megrendült a bizalom a netkereskedelemben, az emberek félnek kiadni bankkártyáik adatait, visszaesik az eBusiness... Törvényeket kell tehát hozni, hogy ez ne így legyen!

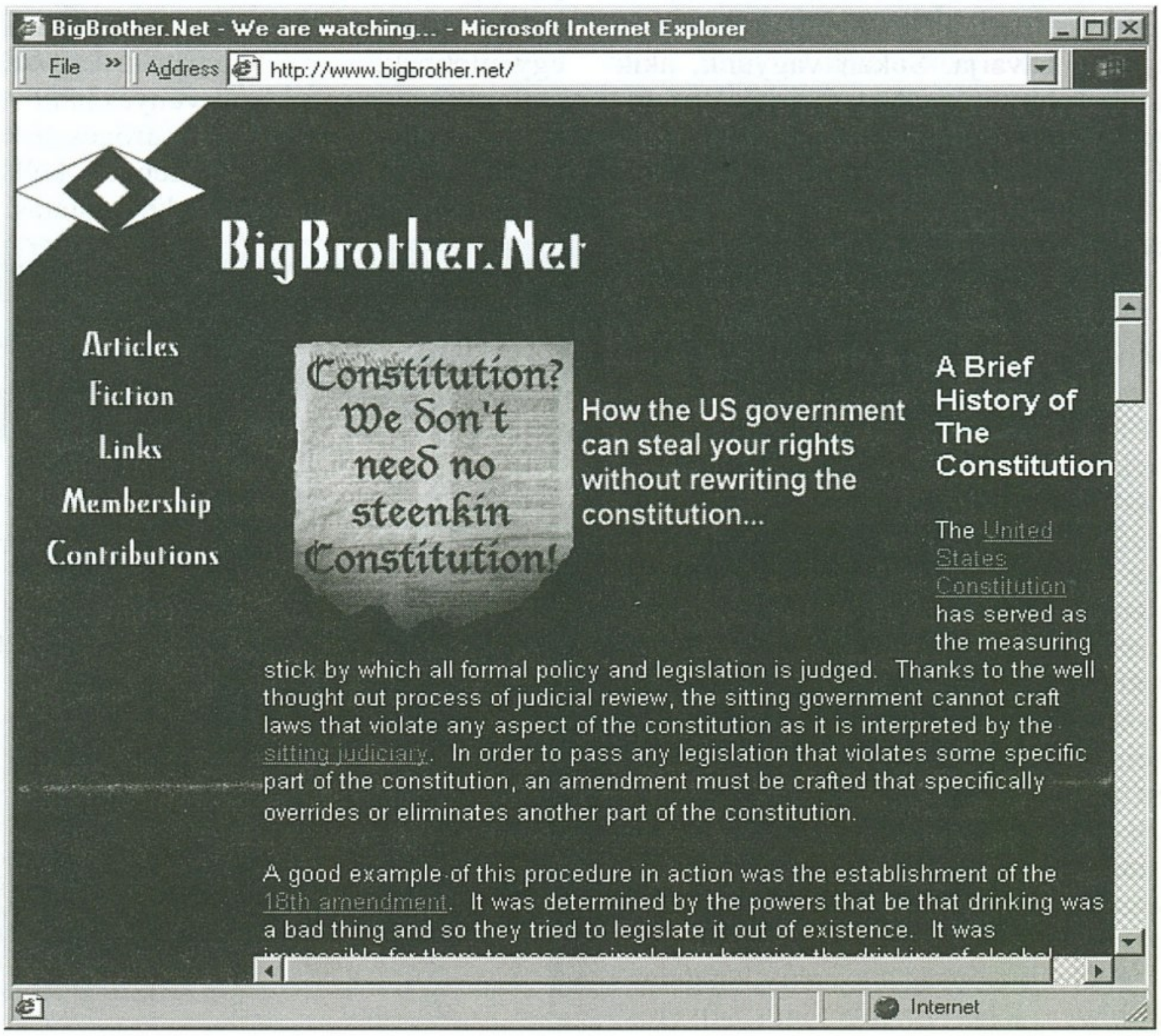
Hmmm. Valóban? A normális gyanakvással megáldott emberek eddig sem adták meg bankkártyájuk számát a Neten át, illetve aki rendszeresen él az e-Kereskedelem lehetőségeivel, az külön kártyát tart, kifejezetten erre a célra, kevés lehívható pénzzel. A Net ugyanis sosem volt az adatbiztonság fellegvára. Az e-Biznisz meg nem tud visszaesni, mert jószerivel még nem is létezik. Most kezd csak feljönni. Az a pár milliárd dollár, amit évente a világon e-B-ben megforgatnak, csak parányi csepp a nagy e-Gészhez képest.

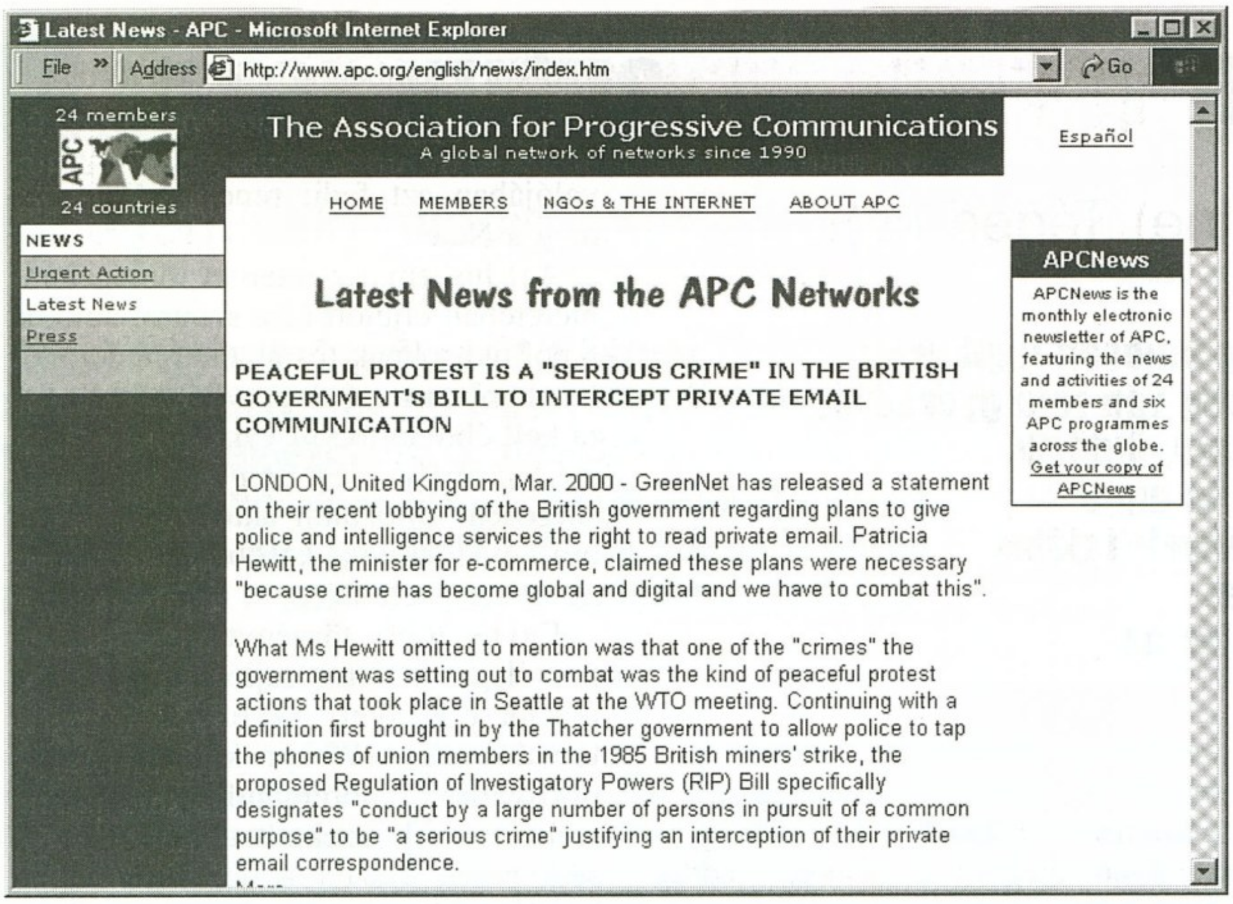
Nekem más gyanúm van, a most újra megélénkült „Védjük meg a Netet" mozgalommal kapcsolatban. Ha jól működik a fejemben az a kis ördögfióka, aki le szokta fordítani a szólamokat tényleges jelentésükre, ez a kifejezés valójában azt fedi: rendszabályozzuk meg a Netet!

Azt hiszem, az internet tömegében, méretében eljutott arra szintre, amikor a politikusoknak (és az őket támogató, hatalomban tartó nagy világcégeknek) rá kell ébredniük: itt van egy hatalmas média, amely (még) nem az övék. Ezen sürgősen változtatni akarnak, mert ha most nem teszik, később esetleg már nem fog sikerülni.

Egyre több törvénytervezet kerül napvilágra a Neten zajló szabad információcsere ellen. Az e-mailre kivetendő adók, a Weben kutakodó „pedofilvadász" kommandók, a szabad webszerverek üzemeltetőinek felelősségi perei, a szerzők jogainak hamis beállítása, kísérletek a titkosító szoftverek eltiltására...

Ezek mind egy irányba mutatnak: a politikusok szeretnék, ha a Net is betagozódna a nyomtatott és sugárzott sajtó rendjébe: túlnyomó többségük legyen a mindenkori hatalom és a tőke szócsöve, kisebb részük kötődhet egy-egy párthoz, pártcsoporthoz a hatalmi viszonyuktól függetlenül, és maradjon a demokrácia demonstrálására minden politikai vonaltól független, a valós történésekről hírt adó néhány média is. Ez utóbbiak miatt érdemes egyáltalán





még újságot venni. Csak ezek vannak a legnehezebb helyzetben. A színes, népszerű magazinok kvázi politikamentesek, eleve más kategóriába tartoznak, de nekik is erős konkurencia lehet a Net, hisz aki netfüggő, az nem sok nyomtatott lapot vesz kézbe, és ritkábban nyúl a pénztárcájába.

A Neten kicsit mások az arányok. Ide mindenki bedobhatja kedvenc nézeteit, rögeszméit; terjesztheti hitét, meggyőződését. A netezőket nehéz „lekapni", azonosítani, felelősségre vonni, sokkal szabadabb az egész. Sokszor szabadosabb is, de aki tud válogatni, azt ez nem igazán zavarja. Sokan vagyunk, akik jobban hiszünk a Neten megjelenő híreknek, mint a hivatalos csatornáknak. Egyrészt azért, mert itt kevesebb a szűrő, másrészt mert szinte minden eseményre lehet valódi szemtanút találni, s így hitelesebb az egész. Egyelőre...

És most jön elő a paranoiám. A rendkívül gyors, jól előkészített, alaposan átgondolt és finoman összehangolt kormányzati reakciók miatt lehetségesnek tartom, hogy a nagy támadássorozat nem egy saját képességeit tesztelő „független" hacker (csapat) ötlete, hanem a törvényi szabályzás igényét világtendenciaként feltüntetni kívánó, hatalmi érdekcsoport megbízásából végrehajtott akció. Azért kell a legnagyobb szervereket (mellesleg rombolásmentesen) „büntetni", mert erre a füstbombára mindenki odafigyel, nagy a sajtóvisszhangja. Kisebb durranásokra — amilyenek naponta előfordulnak — nem lehetett volna politikai akciót szervezni.

A Net biztonságát törvényileg garantálni nem lehet! Ez hibátlan szoftverek, bevált adatbiztonsági eljárások, képzett szakemberek, gondos programozók kérdése. A rendőri netkommandó nem hitelkártyánk védelmét garantálja, hanem a hatalom ellenőrzési jogának kiterjesztését, titkaink kifürkészését... Sokan vannak, akik hisznek az „erős állam = rend = jólét" összefüggésben. Gyanítom, hogy csak addig, amíg a túlkapásoknak mások a szenvedő alanyai. Az állam ne erős legyen, hanem rendes: jogrendes! Olyan, hogy mindenkire, mindenütt és mindig érvényesek legyenek a törvények. És ne legyenek az egyenlőek között néhányan egyenlőbbek!

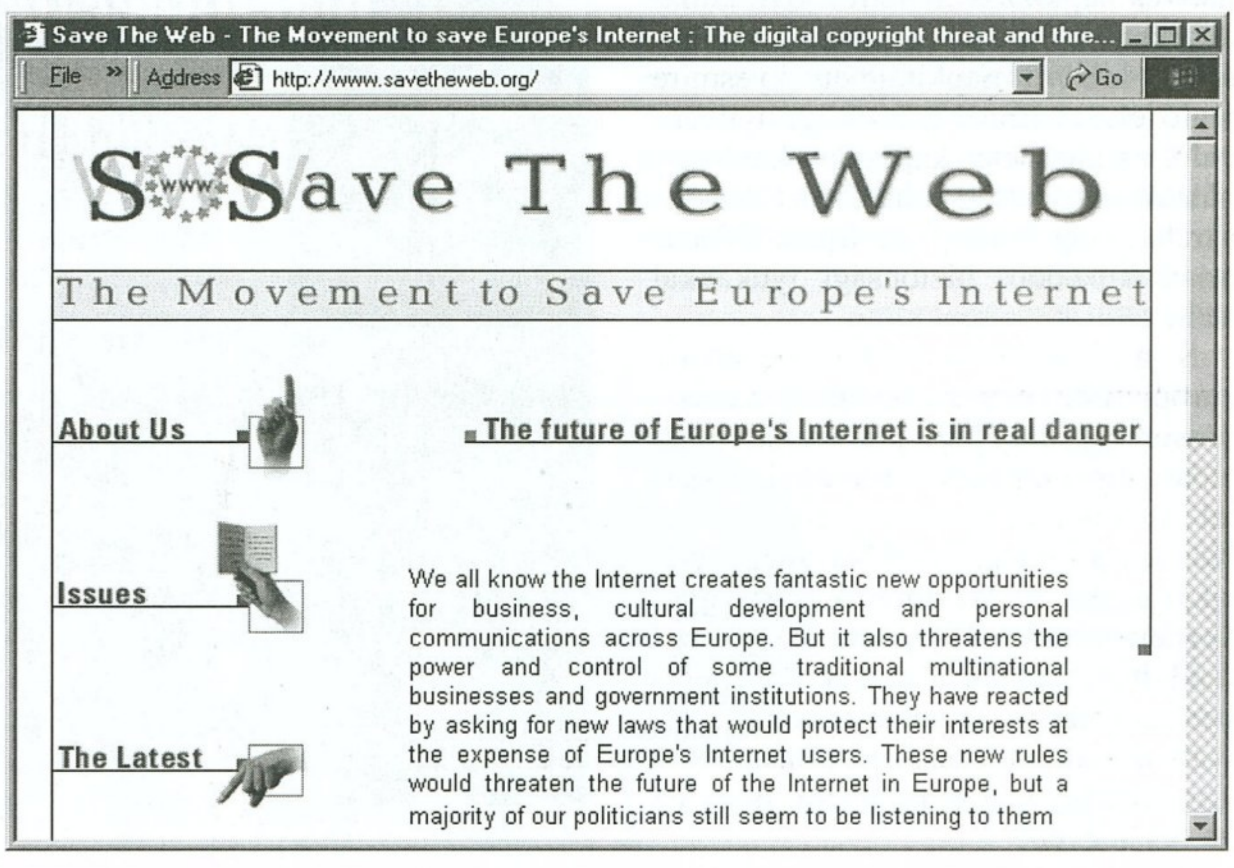
Természetesen a köztörvényes bűnözők (pedofilok, fegyver- és drogcsempészek, terroristák stb.) ellen fel kell lépni, de aránytévesztés és képmutatás ezt a Netre kihegyezni, úgy feltüntetni, mintha a bűnözés jéghegyének láthatatlan, víz alatti részével szemben az igazi veszélyt a látható rész néhány koszos törmeléke jelentené.

A Net a legnagyobb nyilvánosság. Aki annyira szemtelen vagy olyan tudatlan, hogy bűncselekményével kilép a nyilvánosság elé, azt a bűnüldözők inkább baleknak, csalinak, támpontnak használhatják az igazi — jóval kiterjedtebb, de láthatatlan — bűnözői birodalmak felkutatásához.

Ne feledjük, hogy mi magunk vagyunk a Net! Amit felteszünk, azt használják mások, és amit mások közzétesznek, azt olvassuk mi. A műszaki hátteret mi töltjük meg tartalommal. Mi leheljük az életet az egészbe. Ha nem akarjuk, hogy titkaink kiszivárogjanak, ne dobjuk be a nagy közösbe. Ha valamilyen honlap tartalma sérti ízlésünket, ne keressük fel! Ahogy az újságosnál sem kell megvenni a pornólapot, itt sem lehet „véletlenül odatalálni". Ha mégis, akkor tovább is lehet lapozni.

Ha a hackereknek van egy csöppnyi kis eszük, nem csinálnak olyan akciókat, amelyek tálcán kínálják az ürügyet a mindent maguk alá gyűrő hatalmi törekvéseknek a beavatkozásra, a Net megrendszabályozására. Egy operációs rendszer biztonsági hibáinak felfedezése akkor is sikerélmény, ha saját gépük volt a terep. Az eredményt pedig a nyilvánosság előtt tisztességes módszerekkel is lehet demonstrálni. Sajnos tudomásul kell vennünk, hogy a teljes „szabadság" ideje lejárt, a hatalom szemet vetett utolsó, nyílt, szabad játszóterünkre. Rajtunk is múlik: engedjük-e, hogy elvegyék tőlünk.

Sándor Gábor



A telefonszámla nem enged

Egy „kedvezmény” margójára

Mímel: 1: utánoz; 2: színlel, tettet.
MIME: Magyar Internetezők MagánEgylete.
E két szó nem véletlenül került egymás közelébe. Pontosabban az utóbbi egy betűszó, mely eredetileg a Modemes Internetezők MagánEgylete kívánt volna lenni, de az alapítók közben rájöttek, hogy itthon nem csak a modemesek vannak hátrányos helyzetben...

A MIME megalakulásakor Kis János (Johannes) azt a célt tűzte az egyesület elé, hogy képviselje a hazai modemes internetezők érdekeit az internet-szolgáltatók (ISP-k) és a telefontársaságok (elsősorban a Matáv) túlzott profitéhségével szemben. A cél helyes, a megvalósítás módja már kevésbé, mert például az alapszabály tervezete rögtön kirekesztette az ISP-k alkalmazottait a tagok közül. Ez ambivalens dolog, mert egyrészt ugyan tényleg irreális, hogy valaki saját cégével szemben képviseljen bizonyos érdekeket (még ha magánszemélyként őt is érintik a magas tarifák), másrészt viszont elég nagy öngól távol tartani azokat, akik benne élnek abban a környezetben, ezért érdemben tudnának szót emelni, visszásságokat feltárni, és megvalósítható javaslatokat tenni. A csak otthonról internetezőknek általában nincs is olyan rálátásuk a dolgok menetére, hogy egyenrangú tárgyalópartnerek lehessenek a minden részletkérdésben alaposan felkészült hálózatosokkal szemben.

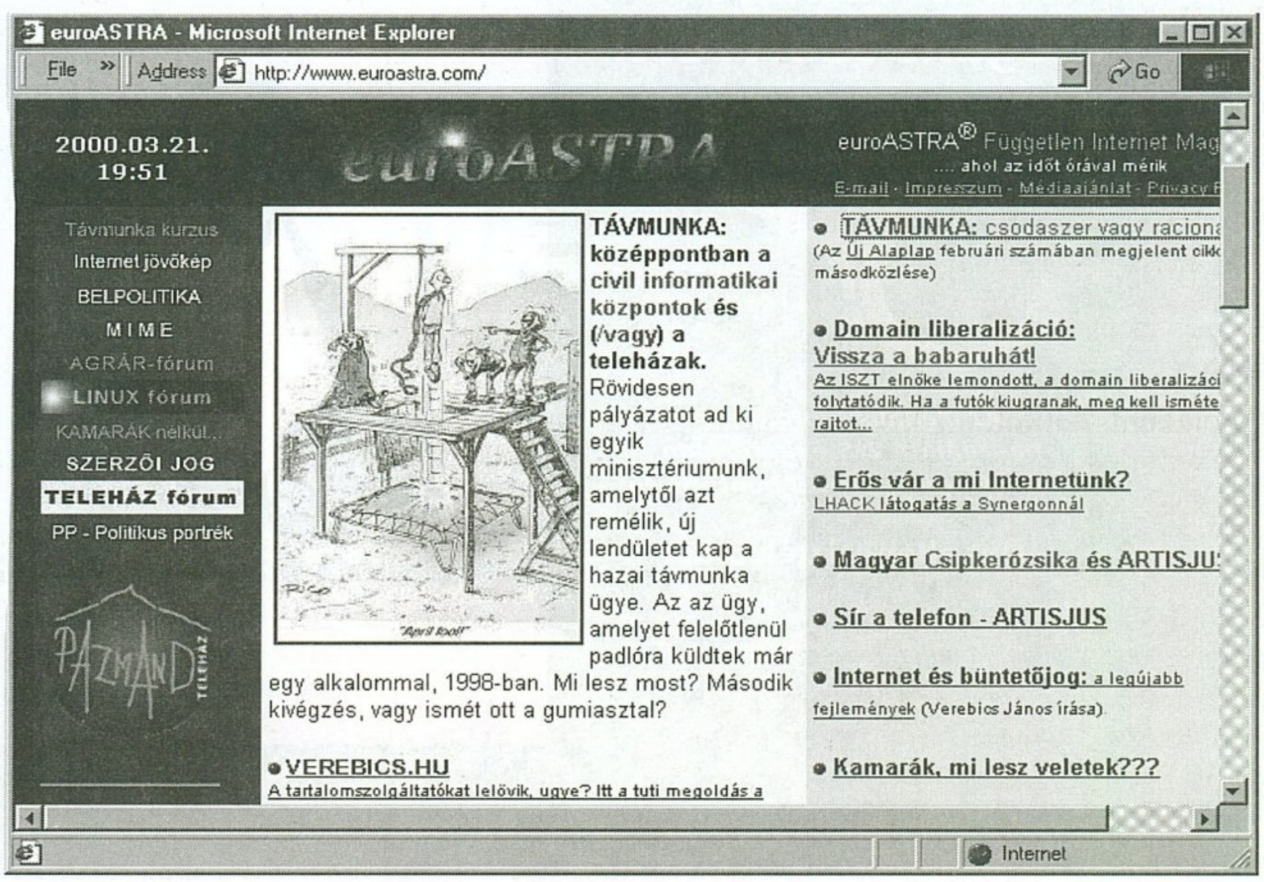
Időközben Johannes lemondott, és a MIME az ügyvivőre, Pósvári Sándorra maradt. Az elnöki tisztet ugyan elvállalta Zorán, aki közismert és tisztelt személyiség, de nehezen tudom elképzelni, hogy helyzete alapján úgy tudna fellépni a jelenlegi Szerzői Jogi Törvény vadhajtásai ellen, hogy szembeforduljon az Artisjusszal és a Zenész Szakszervezetben betöltött elnöki tisztségével. Valószínűleg azt sem kockáztathatja meg, hogy a MIME céljainak megfelelően emeljen szót az SZJT torz értelmezése és elvakult alkalmazása miatt (mennyire tudna például a tavalyi „bilincs vagy pingvin” plakátakció mellett kiállni). A Matávval és más szolgáltatókkal folytatott tárgyalások is nyilvánvalóan az ügyvivőre maradnak, akit ezért tisztelet illet. (Nem kis részben azért is, mert a MIME-tagok a tagdíjat nem nagyon akarják fizetni, lásd erről a www.euroastra.hu lapjain olvasható panaszt.)

A tárgyalások eredményeképpen a Matáv meghirdette a „kedvezményt”: 2000. március 1. után valamikor. Az előfizetésen felül 4-5000 Ft/hó átalánydíjért cserébe a kedvezményes időszakban (hétköznap este 6 és reggel 7, hétvégén és ünnepnapokon délután 3 és reggel 7 között) a hosszasan telefonálóknak nem kell félniük a vonalszakadástól, és éjszakánként akár 20-30 alkalommal is hívhatják számaikat, a számláló nem fog ketyegni.

Tényleg dicsérendő, hogy valami ilyesmi egyáltalán kikerekedett! De itt jön a „mímelés” jelentősége. Kinek az érdekeit sikerült érvényesíteni? Hisz ez a „kedvezmény” csak azok számára előnyös, akik hivatásos internet-használók, és saját bevallásuk szerint is 18-20 ezer Ft-os havi telefonszámlát produkálnak. Nekik ez tényleg óriási engedmény. Más irányból nézve viszont a MIME feladata, hogy minden modemes internet-használó érdekét képviselje, azokét is, akik számára a havi 4-5 ezer forintos telefonszámla elég súlyos korlát. A MIME inkább ellenükre tett, mert segített a Matáv nem igazán szimpatikus piaci manipulációjának elkendőzésében.

Számoljunk csak egy kicsit! Egy átlaghónapban a napi 150 Ft-os Matáv-kedvezmény maximum 30x150, azaz 4500 Ft-os telefonszámlát eredményezhet, ha nem szakadnak meg a vonalak. De sajnos megszakadnak. Ilyenkor az ember újra hív, újra kezdi a 150 Ft-ot, és alkalmanként kipengeti a kapcsolási díjat.

Állítólag technológiailag nem küszöbölhető ki a vonalkapcsolat gyakori bomlása, és ezzel a jelenséggel együtt kell élnünk. Nekem ezzel kapcsolatban vannak ugyan fenntartásaim, de az egyszerűség kedvéért legyen ez a kiinduló tétel. Azt viszont nem fogadom el, hogy a problémát ilyen furcsa átlagolással kell megoldani. A számlázóprogram egyszerű módosításával megoldható volna, hogy azokat a hívásokat, amelyek vonalbontás után néhány percen belül ugyanarra a számra irányulnak, a rendszer este 6 és reggel 7 között ne



kezelje külön hívásként, hanem az előző hívás mérése folytatódjon, illetve amikor elérte a 150 Ft-os határt, ne számlázzon többet ugyanazon hívott szám esetén! Ez nemcsak megoldaná a szakadások problémáját, hanem a „jóságos kedvezmény” meg „engedmény” meg egyéb tetszelgések helyett nevén nevezné a gyereket: „adva van egy kiküszöbölhetetlen hibajelenség, amelynek következménye volt eddig az indokolatlanul beszedett pénz, ezt a jövőben egy kézenfekvő megoldással mindenki megelégedésére kiküszöböljük”.

Nézzük a dolgot más aspektusból is. Sokan vannak (jómagamat is beleértve), akiknek legfeljebb havi 20 órás internetes kapcsolatra futja a családi költségvetéséből. Nekem nagyjából ennyi is az igényem az internet használatára. Főleg levelezek, azt napi 10-20 percben simán lebonyolítom, a nagyobb lélegzetű keresést és letöltést pedig a kedvezményes időszakok elején, egyhuzamban igyekszem megoldani, havi 1-2 alkalommal, így optimalizálni tudom a telefonszámlát.

Aki ilyen 10-20 órás előfizetéssel rendelkezik, az megközelítőleg sem generál 4-5000 Ft-os telefonszámlát, legfeljebb akkor, ha naponta többször, mondjuk 3-5 perces adagokban intézi netezhetnékjét, és ennek egy részét csúcsidőben teszi. Gondolom, ilyen madár aligha van.

Nekünk viszont kedvezmény továbbra sincs. Csak azoknak, akik hosszú vagy korlátlan idejű előfizetéssel rendelkeznek. Tudom, hogy van olyan szolgáltató, amely a kedvezményes időszakban nem számolja az órákat, tehát egy 10 órás account is elnyúlhat havi több száz órára, de nem ez a jellemző. Olyannyira nem, hogy az a cég, amelyik ezt a kedvezményt eddig adta, erősen beszűkítette a „nem számolós” intervallumot, és annak egy részét is gondosan a Matáv telefonkedvezményes időszakán kívülre tette. Ráadásul ebben az esetben megint oda lyukadunk ki, hogy ha nem szakadna meg a vonal, vagy ha a Matáv (és természetesen a többi telefonszolgálató is) normális számlázóprogrammal kiszűrné az ilyen hibákat, akkor nem kellene kedvezménynek titulálni saját hibájuk kijavítását.

Ha egy kicsit még mélyebbre ásunk, további érdekes összefüggéseket fedezhetünk fel. Akinek a netezésből adódó telefonszámlája éppen csak a 4-5000 Ft/hó határán billeg, hogy ebben a sávban megtartsa, folyton lesi az órát, számolgatja a leszakadásokat, a kedvezmény lehetőségét látva most könnyen juthat olyan döntésre, hogy befizeti ezt a fix díjat, és akkor bátrabban netezhet. Úgy érzi, hogy neki ez jól jön... a Matáv viszont tudja, hogy egy bizonytalan számlázással szemben stabil bevételre tesz szert, amit akkor is beszed, ha ügyfele bármilyen körülmény folytán nem használja ki lehetőségeit.

Milyen átalány lenne valódi kedvezmény? A kábelnetes társaságok időkorlát nélküli havi díját, és az ott elérhető sávszélességet alapul véve a telefonos szolgáltatás árának jóval ez alatt kellene elhelyezkednie.

Ha a legdrágább UPC havi 10 000 forint körüli összegre tervezi szolgáltatását, akkor egy telefonos szolgáltatásnak, telefonszámlával, és korlátlan időtartammal legfeljebb a felét volna szabad elkérnie. Vagyis havi 5000 Ft-ot, ami telefonszámla és ISP díj egyben! Ez méltányos lenne. Nem kedvezmény, hanem reálisan a magyar jövedelmekhez igazított szolgáltatás!

Hogyan lehetne ezt megvalósítani? Például úgy, hogy a telefontársaság valami létesítési díjért (legyen mondjuk 10 ezer Ft) kihúz egy vonalat a lakásomba, amelyet csak internet-szolgáltatók hívására használhatok. Ezért kér havi 3000 Ft használati díjat. Figyelem: korlátozott a szolgáltatás, mert csak meghatározott szám(ok) hívható(k) róla modemmel. Kapcsolási díj nincs, forgalmi díj nincs. Ha egész nap használom, akkor is ennyi, és ha be sem lépek hetekig, akkor is. Az ISP pedig ad kb. ugyanennyiért korlátlan elérést plusz egy postaládát. Akinek ennél több kell (honlap, tartománynév, bármi), az fizessen érte külön. A sebesség 28,8 és 56 kbit/sec között lehetne.

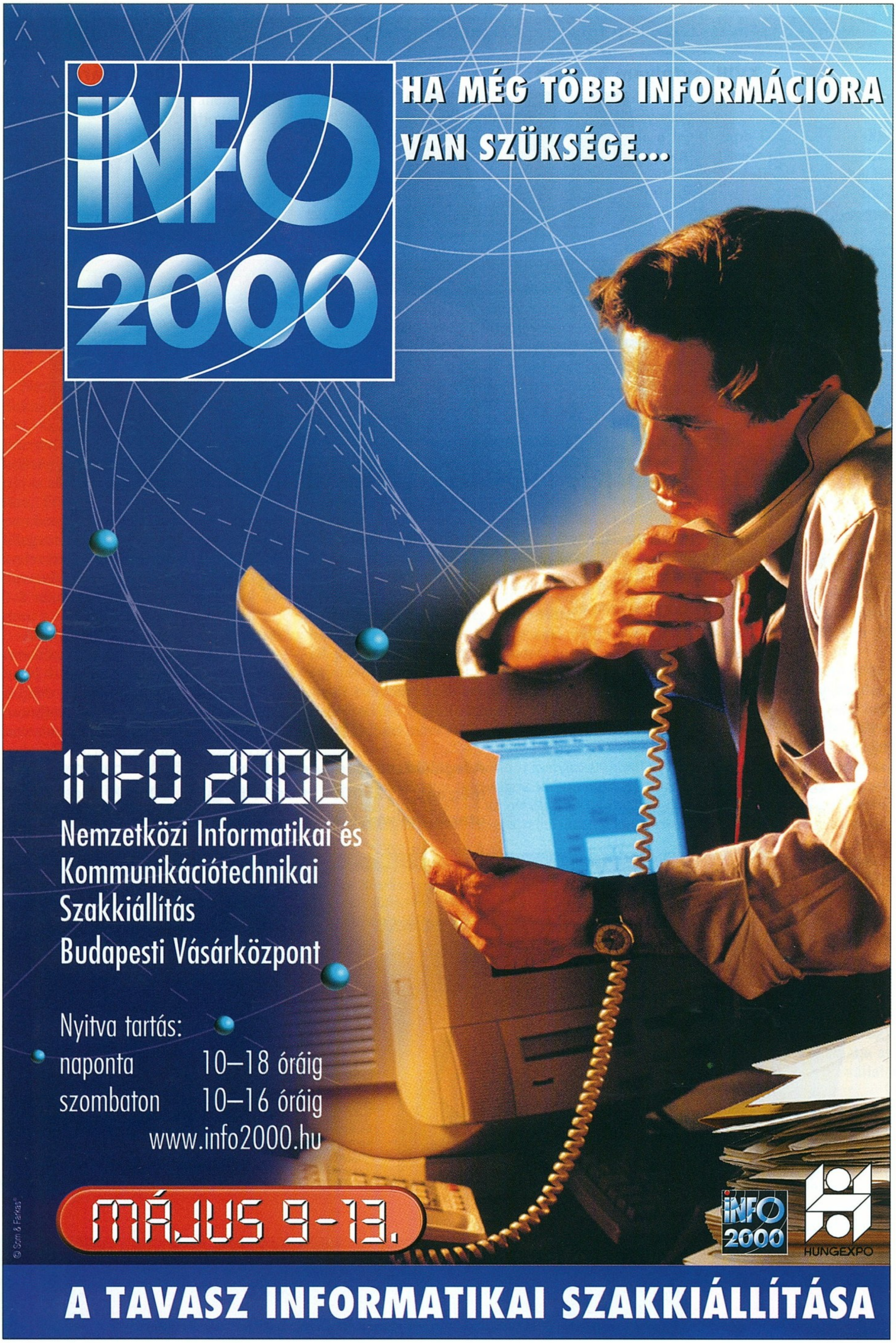
Tudom, hogy ezt a díjat sem képes mindenki megfizetni, de a jelenleginél sokkal többen, tényleg tömegesen tudnának bekapcsolódni a netezésbe. Olyanok, akik számára a mai feltételek mellett ez elérhetetlen. A telefontársaságnál jelentkezne a fejenként alacsonyabb, de összességében több és stabilabb bevétel, az internet-szolgáltatóknál szintén. A nagyobb állandó közönségre már lehetne portálokat, hirdetéseket, áruházakat szervezni... Magyarországon is meglódulna végre ez a piac. Ha a kormányzat komolyan gondolja a felzárkóztatást Európához, esetleg maga is besegíthet a vonalak kiépítésébe, vagy egy ideig támogathatja az ilyen szolgáltatást vállaló cégeket.

Kedves MIME! Ha legközelebb tárgyalóasztalhoz tudod ültetni a telefontársaságokat és az internet-szolgáltatókat, esetleg találva melléjük hosszú távon gondolkodni képes politikust is, akkor lehetőleg már mindannyiunk érdekeit képviseld!



Sándor Gábor
saga@matavnet.hu

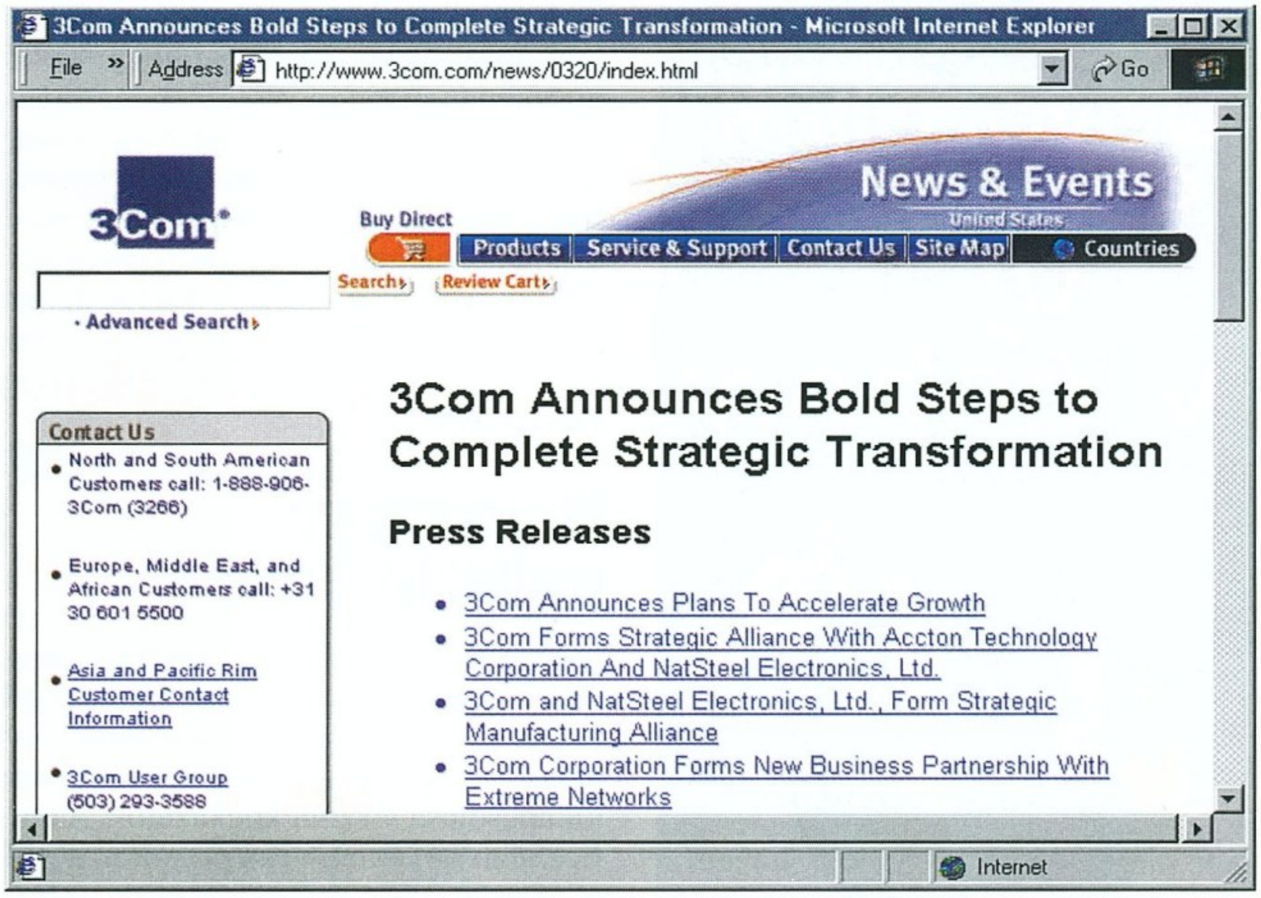
INFORMÁCIÓKÉRÉS: 17 ▼



INFORMÁCIÓKÉRÉS: 14 ▲

A 3Com profilváltása

Sokáig úgy tűnt, hogy a 3Com neve elválaszthatatlan a lokális és nagyobb kiterjedésű vállalati hálózatoktól, a hozzájuk kapcsolódó eszközök gyártásától. Amikor először bukkantak fel a „profilidegen”, inkább a szórakoztató elektronika felé kacsintgató 3Com termékek, még amolyan „félrelépésnek”, kikapcsolódásnak tűnt az egész. Mostanra bebizonyosodott, hogy a régi termékvonaltól végleg el akarnak szakadni. A LAN/WAN üzletág mostani leépítése mintegy 3000 alkalmazottat érint. Kétharmaduk átmegy a 3Com partnercégeihez, egyharmaduk közös megegyezéssel kilép. Ez év közepén abbahagyják többek között a CoreBuilder L2/3 Gigabit Ethernet és ATM LAN kapcsolók, illetve a PathBuilder és NetBuilder kapcsolók és routerek gyártását. Emellett egy részben általuk alapított új cég veszi át az internetes eszközök kutatását, tervezését és forgalmazását, beleértve a US Robotics márkájú analóg modemeket is. Irány tehát a könnyű műfaj, amit csinálni persze szintén nem könnyű, de a 3Com a jelek szerint mégis ebben látja a jövőjét, s feladja a „komoly zenét”. További információk olvashatók a http://www.3com.com/news/0320 /index.html weblapon.



Elender-betörés, 3. felvonás

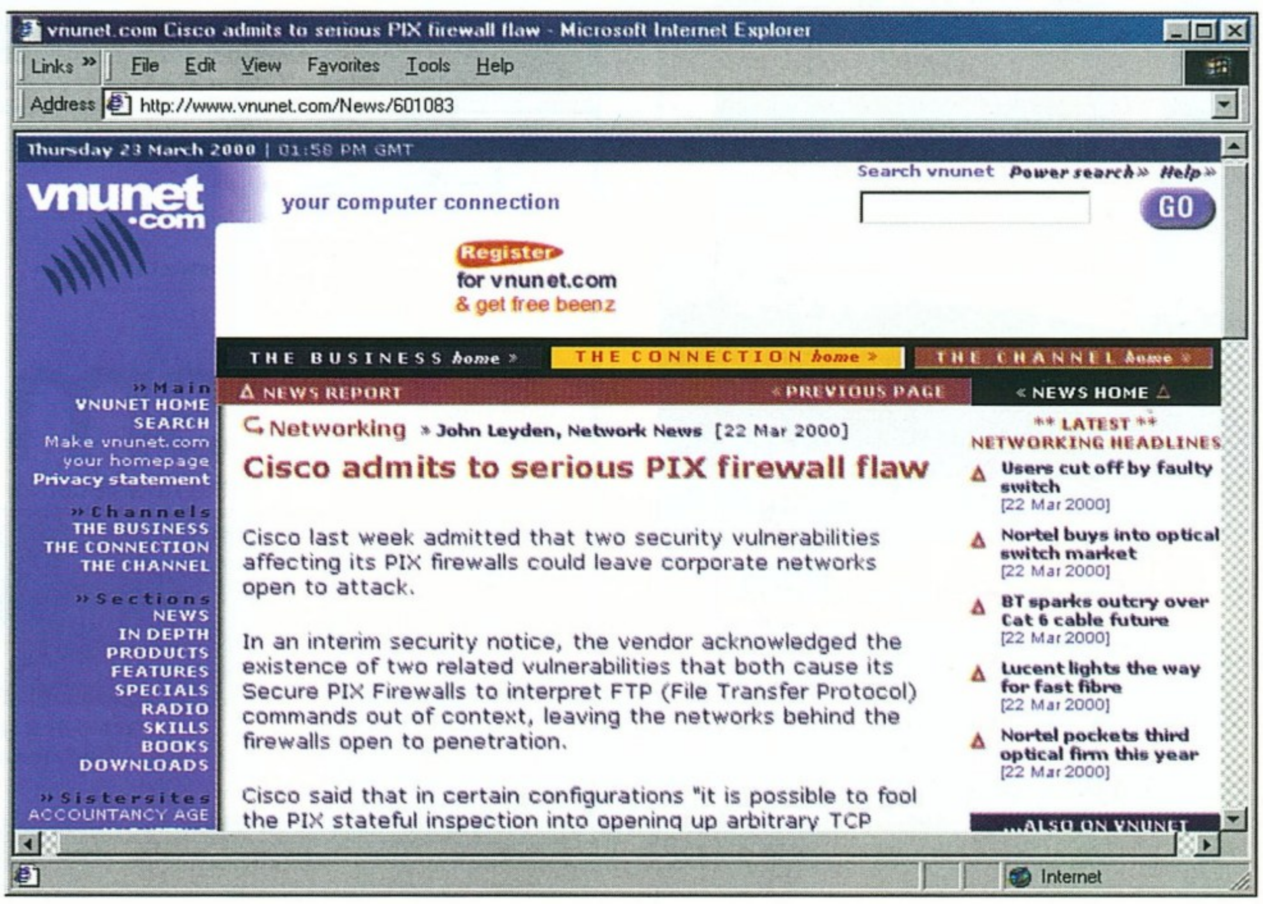
Három egymás utáni számunkban is téma lett az Elender rendszerébe történt betörés, de ezúttal már a felderített tetteseknél tartunk. Megdőltek azok a feltételezések, hogy személyes bosszúhadjárat volt az akciók mögött. Az árulkodó nyomok és az elszemtelenedő ismételt betörési kísérletek alapján jutott el végül a nyomozás négy 15 és 19 év közötti fiúhoz, akik hálózati tudásukat autodidakta módon szerezték meg, a betörésre viszont profi módon használták. Minden eddigi jel szerint a behatolásokat pusztán „brahiból” követték el. Ennek ellenére büntetésük akár 5 év börtön is lehet, tekintettel a több tízezer internet-használónak és az Elendernek okozott kárra. Életkoruk és enyhítő körülményeik miatt mégis kiszámíthatatlan, hogy végülis milyen következménye lesz ténykedésüknek.

„Sláger” a biztonság

A nagy külföldi webszolgáltatók működését lebénító akciók, a hálózaton át terjesztett vírusok, a rendszerekbe történő behatolások, a szoftverek biztonsági rései időszerűvé tették, hogy a biztonsággal intenzívebben foglalkozzunk. Február végén az IDG Magyarország szervezett konferenciát a hálózatok biztonságáról. A Kürt, az ORFK, a Telnet, az InNet, az Icon, a Microsoft, a Novell, a HP és az SCO képviselői egyrészt az adatok biztonsága, másrészt a rendszerek és hálózatok üzemképessége oldaláról elemezték a témát. Az utóbbi különösen az internet és az erre alapozott elektronikus kereskedelem szempontjából kulcskérdés. Ezen a konferencián is felszínre került azonban, hogy bár a számítástechnikai rendszereket kívülről fenyegető veszélyek és a látványos betörések kapnak nagy nyilvánosságot, a rendszerek elleni manipulációk többségét belső munkatársak követik el, sőt a külső behatolók is gyakran támaszkodnak belső közreműködőkre. Mivel a hálózati károkozás igen nagy része felderítetlen marad, a pontos arányokat nehéz lenne megállapítani.

Lyuk a tűzfalon

A VnuNet információi szerint a Cisco is elismerte, hogy a PIX tűzfalak nem a kívánalmaknak megfelelően funkcionálnak. Pontosabban a Cisco Secure PIX Firewall termékek a 4.2(5), 4.4(4) és 5.0(3) verziójú szoftverekkel az FTP (File Transfer Protocol) használatakor kinyitott kapukat nyitva is hagyja, ezzel



szabad utat enged az illetéktelen behatoláshoz. A hír kiemeli, hogy ez a probléma éppen a Cisco nagy piaci súlya és termékeinek elterjedtsége miatt jelent komoly problémát, különösen, hogy tűzfal termékei is népszerűek. Egy másik probléma, hogy a „becsomagolt" FTP parancsokat a Cisco tűzfal hajlamos önálló programként futtani, ami szintén lehetőséget nyújt külön átjárók kialakítására.



Vírusvédelem mellékhatásokkal

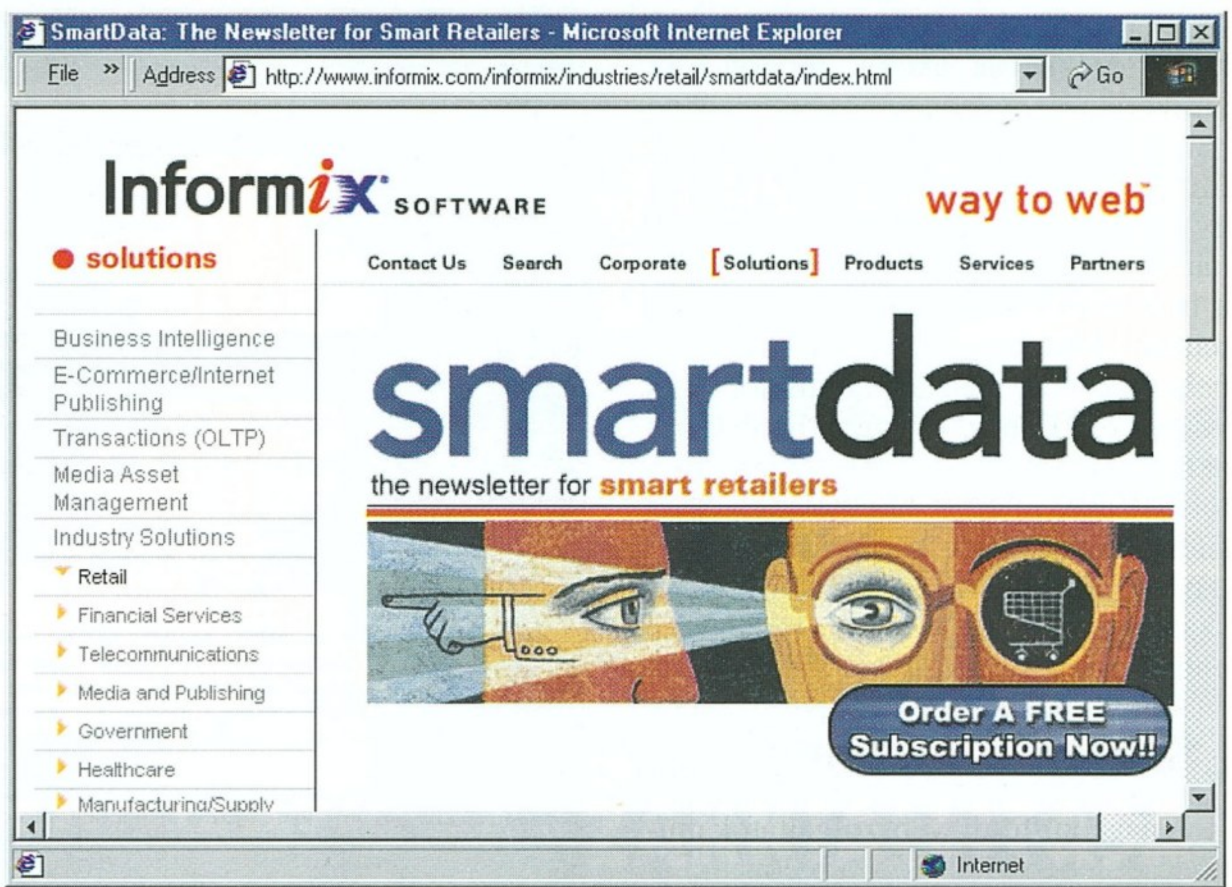
Egyes antivírus programok a számítógép üzenetfeldolgozó és fájlkezelő rendszerébe beülve kellemetlen mellékhatásként lassítják a munkát. A figyelés kikapcsolása viszont kockázatos. A OfficeScan nevű antivírus program, mely a „Denial of Service" (DoS) típusú „géplebénítást" is szűrni hivatott, a Bugtraq és a szoftvert készítő Trend Micro közlése szerint olyan hibát tartalmaz, amely teret enged a belső támadásoknak (internal attack), mert a kliensek azonosítását nem oldja meg. Az ügy pikantériája, hogy a Microsoft kínálatában is szerepel az OfficeScan for Microsoft Small Business Server 4.5 ingyenes példánya a 2000. március 1-től június 30-ig terjedő vásárlási akcióban.

Az Allied Telesyn piaca

A japán központú Allied Telesyn International március végén sajtókonferenciát hívott össze a CeBIT utáni helyzet ismertetésére. Ezt a cég piacpolitikájában bekövetkezett szemléletváltás indokolta, térségünket pedig árérzékenységünk okán érintheti. A világranglistán ötödik helyezett hálózati eszközszállító a kis- és középvállalati szegmensben még előkelőbb helyen, a legkisebb és az otthoni hálózatok piacán pedig az első helyen végzett 1999-ben. A cég európai központjának munkatársai jó esélyt látnak rá, hogy ezt meg is tudják tartani. Térségünkben ugyanis megvan az igény a magas technikai színvonalat képviselő, ugyanakkor a kisebb hálózatok kiépítéséhez is költséghatékonyan felhasználható eszközökre. Az utóbbi években a szoftvereszközeiken végrehajtott fejlesztések eredményeként támogatják a hálózatos ipar nyílt szabványait, és akár a webböngésző által nyújtott felületen is elvégezhetők a szükséges hálózati beállítások.

Informix bemutató

Nemrég mutatták be az Informix új termékpalettáját Magyarországon. Az adatbáziskezelő rendszereket fejlesztő Informix legújabb csomagja az Informix Internet Foundation (IIF 2000), mely különböző platformokon, Unix,



Linux és Windows NT alapú hálózatokon kínál megoldást a nagyvállalati adatkezelési feladatok megoldására. Az Enteprise Java Beans és az ActiveX alapú COM komponensek a szerveren futva biztosítják a válaszidő csökkentését, az interneten a teljes értékű adatbáziselérést IIF rendszer alatt.

Az IDS 2000 (Informix Dynamic Server) lehetővé teszi a tisztán relációs adatbáziskezelést, és az objektumorientált megközelítésen alapuló megoldásokat is. Az adatbáziskezelő belső átalakítása és több új változótípus bevezetése révén lényegesen bővültek az adattárolás lehetőségei, például eltérő adattípusok, nagyméretű szövegek és multimédiás adatok integrált kezelésével. A csatlakozó fejlesztések érdekében az IDS 2000-nek része a Datablade Developer Kit is.

Compaq, Win2k, Unisys

A Compaq többféle Unix alapú operációs rendszert telepít gépeire, de a Win2k OEM mellett szintén az elsők között kötelezte el magát. Egy új megállapodás értelmében a Unisys által 32 bites Intel Pentium III Xeon és 64 bites Itanium processzorokhoz kidolgozott CMP (Cellular MultiProcessing) architektúrát használják fel a Windows 2000 Datacenter Serverre optimalizálva. A Compaq ProLiant márkanéven fogja szállítani az új 32 processzoros rendszereket. A megállapodás kétéves időtartama alatt ez mintegy 400 millió dolláros potenciális bevételt jelent a Unisys számára.

Oktatási adomány

A Hewlett-Packard és az IBM öszszesen 100 millió forint nagyságrendű összegű termékkel támogatja a Budapesti Műszaki és Gazdaságtudományi Egyetemen folyó oktatást, főleg a tervezési munkákkal kapcsolatos számítások, modellezések területén.

Együttműködő IBM

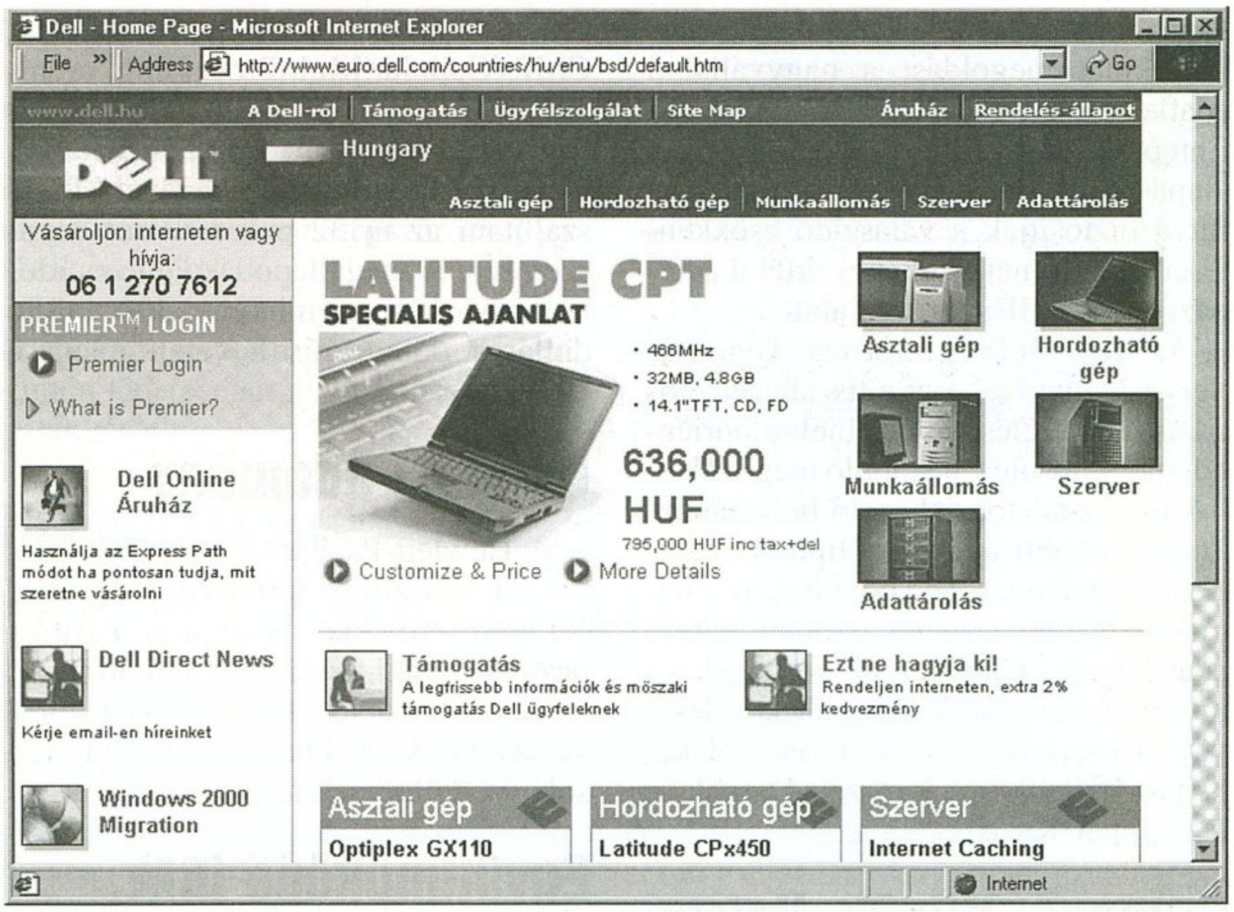
Az IBM korábbi Windows NT-s szoftvereit átdolgozta Windows 2000 platformra. A mintegy 300 termék között találhatók fejlesztőeszközök éppúgy, mint csoportmunka szoftverek. A többé-kevésbé továbbra is önálló életet élő Lotus termékeket is felkészítették a Win2k várható elterjedésére. Más operációs rendszerek használóinak szintén rendelkezésre áll számos IBM termék. A Novell NetWare 5.1 doboza tartalmazza a webalkalmazások fejlesztési eszközeit, az IBM WebSphere család tagjait, melyekkel a beépített Java motor lehetőségei jobban kihasználhatók. Linux platformra is nagy intenzitással tart a fejlesztés az IBM-nél, és lehet, hogy a nagy rendszerek irányából közeledő cég számára ez a legkedvesebb feladat. A Linux mélyreható támogatását jelzi a fájlrendszer szintű fejlesztésekbe való bekapcsolódás. A LinuxWorld-ön jelentették be az IBM részvételét a nagy terheléskor is gyors és hibatűrő fájlrendszerek fejlesztésében (Journaling File System — JFS). A fejlesztés helyzetéről és a szabadon elérhető kódokról további információk is megtalálhatók a http://www.ibm.com /developerworks címen. Az utóbbi idők linuxos IBM-hírei közé tartozik még, hogy bejelentették a NetObjects TopPage for Linux béta verzióját, a ViaVoice megjelenését Linuxra, és hogy a Network Station Series 2200 és 2800 kliensek már szintén futnak Linux platformon.

Ellátási láncok

Az elektronikus kereskedelem cégek közötti változatában (business to business, B2B) az ellátási láncok módot nyújtanak a raktárkészletek radikális csökkentésére és a rugalmasabb piaci alkalmazkodásra. Mivel a sok telephellyel rendelkező cégek ellátásában ezeknek a láncoknak megnő a jelentősége, gyakrabban találkozunk majd az eredeti „supply chain" angol kifejezéssel. Az informatikai hátteret nyújtó cégek közül a közép-európai piacon való jelenlét megerősítése érdekében nemrégiben például megállapodást kötött a Manugistics és a United Transfer. A piacon megjelenő csomaghoz a Manugistics adja azokat az internetes megoldásokat, melyek a világ más területein már beváltak az elektronikus kereskedelemben és az ellátási láncok menedzselésében. Az informatikai rendszerek kialakításával és szaktanácsadással foglalkozó United Transfer pozíciói pedig a piacépítésben erősek, nagy az ügyfélköre ebben a térségben (Coca-Cola, Suchard, Bestfoods, ABN-Amro, Nestlé, United Biscuits, Cofinec, Pannon GSM stb.).

Adatbányász OLE

Windows platformon az adatkezelést kibővítő OLE DB jól használható az adatbázisokkal végzett munkában. A nagy adatrendszereknek vannak azonban olyan sajátosságai, melyeket a korábbi specifikációk nem támogattak megfelelő hatékonysággal, miközben magának a Windows platformnak, illetve az erre épülő SQL adatbáziskezelőnek a teljesítménye folyamatosan nőtt, és a Windows 2000-hez már készítik a kifejezetten adatraktárakra tervezett változatot. A nagyobb igényeknek és lehetőségeknek kíván eleget tenni az OLE DB for Data Mining, amely az SQL nyelven alapuló szabványosítás mellett megtartja az OLE DB előnyeit. Az új adatkezelési protokollt 1999 elején mutatták be, és azt a Microsoft azóta folyamatosan fejleszti. Ennek keretében igyekeznek megoldani a Data Mining Group (http://www.dmg.org) által igényelt kompatibilitást a Predictive Model Markup Language (PMML) leírónyelvvel. Az OLE DB-ről, illetve az új termék béta verziójáról további információk a http://www.microsoft.com /data/oledb oldalon találhatók.



Itt az online Dell

A Dell közvetlen értékesítési stratégiájával és az elektronikus kereskedelemben alkalmazott megoldásaival már többször foglalkoztunk lapunkban. 2000. márciusától itthonról is működik a online rendelés. A Dell a magyar ügyfelek számára megnyitotta a http://www.dell.hu internetes honlapot. Fenntartója a Dell számítógépek forgalmazásában eddig is kiemelt szerepet betöltő HumanSoft. Az interneten rendelt eszközöket ők szállítják, és további támogatást nyújtanak a termékekhez. (Honlapjuk honosítása lapzártánkig még nem volt tökéletes, felváltva szerepeltek benne az angol és a magyar nyelvű szövegblokkok, de dolgoznak a két nyelvi variáns szétválasztásán.)

Újdonságok Xerox-tól

A Xerox a számítástechnika, az informatika legrégibb hagyományokkal rendelkező és kiemelkedő teljesítményt felmutató vállalatainak egyike. Fejlesztési eredményeivel állandóan találkozunk, elég ránézni az egérre vagy a képernyőn a grafikus kezelői felületre. A gyakran technológiai áttörést jelentő fejlesztőmunka napjainkban is zajlik, az utóbbi időben főleg a digitális dokumentumkezelés területén. (Legutóbb elektronikus papírjuk keltett nagy szenzációt, lásd Új Alaplap, 1999. szeptember.) Nemrégiben a Xerox hazánkban is bemutatta a FlowPort rendszert, mellyel a papírdokumentumok kezelésének hatékonysága növelhető. A digitalizált anyagot a központi dokumentumtárból ugyanis bármikor lehívhatjuk. A dokumentumhoz saját fedőlap tartozik, hogy bejelölhessük, milyen további munkafolyamatok esedékesek. Instrukcióink és a dokumentumlapokra kerülő azonosító segítségével a FlowPort „intézkedik" az anyag kinyomtatásáról, elfaxolásáról, levélként való elküldéséről. A dokumentumtárolás integrálható más rendszerekkel is (például a Lotus-szal). A nyomtatók közül bemutatták a Document Centre berendezések újabb típusait. Ezek másolásra, szkennelésre is alkalmas, nagy teljesítményű lézernyomtatók, melyek 60-tól 200 gramm/m^2 súlyú papírra nyomtathatnak, mert a lapoknak csak 17 cm-es pályán kell végighaladniuk.

Olvasni egyre több nyelven

Recognita Plus 5.0 — néhány mínusszal

Nagy érdeklődéssel vettem kézbe a Recognita OCR programjának legújabb verzióját, hiszen a rendszer fejlesztését a 2.0 változat óta kísérem figyelemmel, és az újabb kiadásokban mindig találtam további jó tulajdonságokat, de felesleges és pontatlan dolgokat is.

A telepítés első nekifutásra gyorsan és hibátlanul lezajlott. Nálam a kiválasztott magyar és angol szótárakkal a telepítés 40,7 MB-ot igényelt. Az egyedi telepítéssel szimpatikusnak tűnik, de kevésbé rugalmas, mint amennyire szükséges lenne. A lehetőségek:

— Programüzenetek nyelve.

— A Súgó nyelve.

— Szövegkonvertálók.

— Szótárak.

A legmeglepőbb jelenség a telepítőprogram működése során akkor ért, amikor pár órával később újra aktiváltam a CD-n lévő telepítőt. Csupán azért indítottam újra, hogy elolvassam az egyedi telepítési menü teljes választékát. Annak ellenére, hogy változtatási parancs nélkül távoztam, a teljes telepített Recognita könyvtár eltűnt a gép winchesteréről, így a telepítést meg kellett ismételnem.

Feleslegek

A telepítés befejezése után megnéztem, mit is pakolt fel a gépre, és bizony leszedtem azokat, amelyekre nincs szükségem. Ezek:

— A programhoz kiadtak PDF formátumú dokumentációkat, ezért felkerült a gépre az Acrobat Reader 3.01 telepítőcsomagja (4 MB.) A gépen azonban már telepítve volt a 3.02-es változat. Ennek ellenére kérdezés nélkül berakta a régebbit a Recognita könyvtárába. Nyugodtan lehetne ez is az egyedi telepítés egyik választható komponense.

— A programrendszer egyik új eleme a Caere Scan Manager 5.0. Ez a komponens a rendszerhez csatlakoztatott szkennerek és kamerák hatékonyabb támogatását hivatott szolgálni, és teljesen automatikusan települ (2,1 MB), ráadásul az OCR programtól független helyre (C:\Program Files\Common File\Caere). Meggyőződtem róla, hogy az általam használt Genius Vivid Pro II típusú szkenner nem szerepel a támogatott eszközök listáján, és a Scan Manager ellenére a Genius szkenner saját gyári Twain vezérlőjével működik, teljesen korrektül, így a másik szkennerprogramot levettem (uninstall + kézi törlés) a rá vonatkozó dokumentumokkal együtt (SCANSETUP_E.HTM és SCAN_HUN.PDF — 2 MB).

— Szintén új eszköz a csomagban a SearchRCD nevű segédprogram, amely a Recognita dokumentumok (*.RCD) közötti keresést hivatott megkönnyíteni. Ez is betelepedett a Recognita OCR könyvtárába, kérdezés nélkül. Hasznos lehet a munka során, nem is túl nagy (329 KB), ezért fent hagytam, de a telepítő itt is megkérdezhetné, hogy kívánjuk-e.

— Végül a telepítő hatáskörébe tartozhatnának a tesztképek is. Ezekből automatikusan települt egy TIF formátumú fekete-fehér kép és egy JPG formátumú színes kép. A kettő helyigénye 2,7 MB. Miután megnéztem őket, töröltem az OCR könyvtárából.

A telepítővel az a legnagyobb bajom, hogy nem támogatja a komponensek egyenkénti telepítését és törlését (ahogy például a Word megteszi), hanem az új beállításnak megfelelően minden alkalommal újratelepíti az egész rendszert.

Főbb újdonságok

Cirill karakterek. A program képes felismerni a cirill ábécé karaktereit. Ennek köszönhetően támogatja az alábbi nyelveken írt dokumentumok felismerését: bolgár, belorusz, csecsen, kabard, macedón, moldvai, oszét, orosz, szerb, ukrán. Ezekkel együtt a rendszer már 114 nyelven képes dolgozni.

Saját nyelvlista. Mivel a rendszer ilyen sok nyelven tud, de senki nem használja egyidejűleg mindet, ezért egyedi nyelvlistát állíthatunk össze.

Színes dokumentumok. Színes forrásképek is tárolhatók, megjeleníthetők és kinyomtathatók. Vagyis a Recognita minden eszköze támogatja a színes anyagokkal végzett munkát. Szerintem ez a funkció azzal válhatna teljessé, hogy a dokumentumkép betűinek színe alapján a felismert szöveg betűi is színesek legyenek, ha a felhasználó kéri. (Persze elég lenne 16 színre redukálva.)

A program (a dokumentáció szerint) oldalferdeséget korrigáló algoritmust kapott, de már a Recognita 4-es változata is képes 12 foknyi ferdeség korrigálására, és tapasztalataim szerint az 5-ös változat sem korrigál ennél nagyobb eltérést.

Szövegellenőrzés

A rendszer 4.0 változatában kétféle szövegellenőrző eszköz állt rendelkezésünkre.

1. Szóképes. A felismert szövegben egérkattintással kiválasztott szó grafikus képét a kurzor fölött megjeleníti egy keretben, és az a következő műveletig látható. Ez a funkció változatlanul átkerült az 5.0-s verzióba is.

2. Szövegképes. A felismert szövegben a kurzor által meghatározott helyen lévő karakter grafikus képét folyamatosan követhetjük a képtérfélen, kék határolóvonallal körülvett karakter formájában. A funkció a műveletek közben folyamatosan aktív marad, míg azt ki nem kapcsoljuk. Ez a funkció képtérféli ellenőrző néven került át az új programba.

3. Dinamikus ellenőrző. Ez új funkció a rendszerben, tulajdonképpen a szóképes ellenőrző módosított változata, és abban tér el elődjétől, hogy képes folyamatosan követni a kurzor pozícióját. Tartalma látható marad mindaddig, míg az Esc gombbal fel nem függesztjük, vagy ki nem kapcsoljuk az ellenőrző használatát.

Új felismerési módszer

A 4-es változatban a meglévő 5 felismerési módszer mellé beépítettek egy hatodikat is. Ez 11 nyelv esetében alkalmazható. Működésének lényege, hogy a forrásanyagot egyidejűleg két eltérő felismerési algoritmus dolgozza fel, mindkettő szótárakat is használ, majd az elemzés végén összehasonlítja a két eljárás eredményét. Így javul a felismerés pontossága. Ez a módszer csak akkor használható, ha a felismerendő nyelveket az eljárás támogatja,

továbbá nincsenek megadva egyedileg felismerendő karakterek. Az új módszer által támogatott nyelvek: angol, dán, finn, francia, holland, német, norvég, olasz, portugál, spanyol, svéd. (Sajnos a magyar nincs köztük, pedig magyar fejlesztésű szoftvertől ez elvárható volna.)

A Braille-írás

Jelentős komplex fejlesztés a programban. A feladat megoldása rendkívül sok buktatót tartalmaz, hiszen egy valódi Braille-ábécével nyomtatott könyv lapjai teljesen fehérek, a domborulatok felismerése pedig a szkennerek számára nehéz. Némi kontraszt- és fényerőmanipuláció segíthet ugyan, de az azonosítás esélye még így is nagyon csekély. További probléma, hogy egy Braille-írású könyv lapjait kétoldalasan nyomják, és az a pötty, amely a papír egyik oldalán domborulat, az ellentétes oldalán homorulat. A kétféle kiterjedés egymástól való szeparálása a szkenner számára megoldhatatlan, enélkül pedig a pontmátrixok felismerése lehetetlen.

Egy Braille-ábécé segítségével és szövegszerkesztővel megrajzolt pontmátrixokkal megpróbáltam megállapítani, mi történik, ha a domborulatokat szabályosan elhelyezett fekete pontokkal helyettesítem. A kísérlet eredménye az, hogy így a felismerés sikeres lehetne, ám még mindig maradnak problémák. Nevezetesen a Braille mátrixok egymástól való vertikális és horizontális távolsága gyakran félrevezeti a programot. Azért teszi, mert a szomszédos karaktermátrixok pontjait nem tudja elhatárolni, és összekombinál olyan pontokat is, melyek nem összetartozóak. Ide kapcsolódó probléma, hogy a méretezés lehetősége ellenére bizonyos esetekben a rendszer nem tesz különbséget a pontok jobb és bal oldali pozíciója között. (A Braille-ábécét a mellékelt ábra mutatja, forrás: Vakok Általános Iskolája, Budapest.)

Az táblázatot tanulmányozva láthatjuk az alábbiakat:

— A karakterek pontmátrixai 2x3 pontból állnak.

— A kettősbetűket (mint például CS, GY, TY) egyetlen mátrixban ábrázolják.

— A számjegyek karaktereit több mátrixon ábrázolják, és az első mátrix minden számjegynél ugyanaz.

— Érdekes jellegzetesség, hogy a grafikailag hasonló betűk mátrix formája között szintén kicsi az eltérés, ami felismerési hibát okoz. (A-Á, E-É, I-Í, O-Ó, vagy akár az L és az LY, de még számos betűpár.)

Néhány apró észrevétel

— Bizonyos műveleteket eddig csak egérrel lehetett elvégezni, most azonban billentyűkombinációkhoz is hozzárendelték azokat.

— Ha a Recognita Plus programot a /blind parancssori paraméterrel indítjuk el, akkor az egyébként grafikusan megjelenő szabályozási funkciók néhány esetben szöveges formát öltenek, így azokat a karakteres felolvasó és hangszintetizáló rendszer kezelni tudja.

— A táblázatokban használandó tizedesjel szimbólum kiválasztható. A táblázatok kezelése a 4.0 verzióban jelent meg. Ennek egyik újdonsága: az Eszközök/Opciók/Nézet/Szimbólumok /Tizedesjel menüben a választási listán háromféle szimbólum közül választhatunk (pont, vessző, kettőspont).

— A sablon, amely egy oldal zónaszerkezetét határozza meg, kétoldalas is lehet, így könnyebben dolgozhatunk fel kétoldalas nyomtatványokat, könyveket.

— A dokumentáció szerint az új verzióban tökéletesítették a Word 97-es fájlok exportálását, de nem részletezik, hogy ez miben nyilvánul meg.

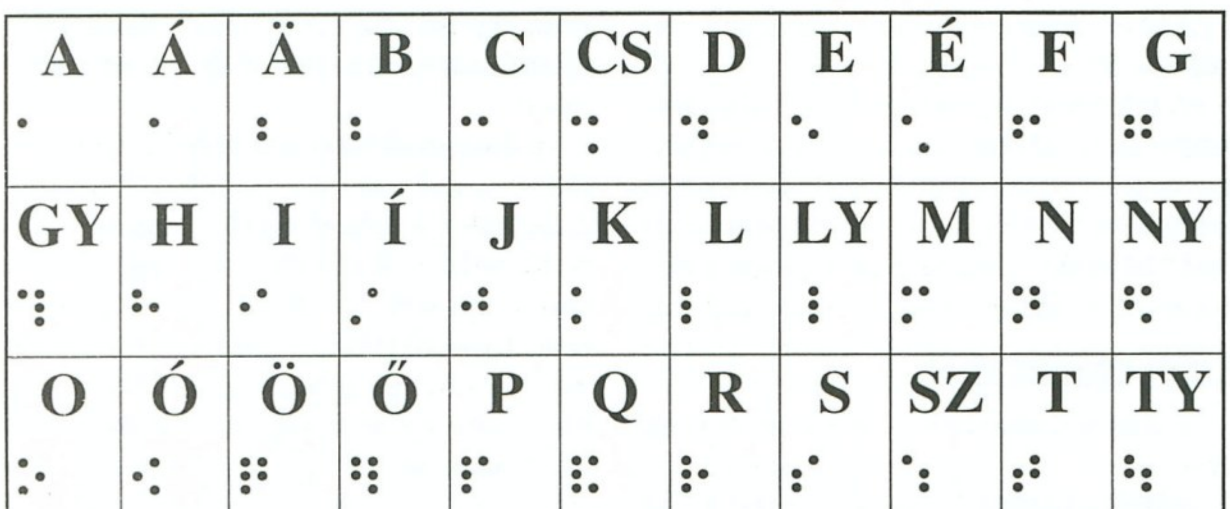
— A rendszer oldalböngésző ablaka a 4-es változat újdonsága volt. Elfogadtam, mert többoldalas anyagok kezelésekor néha jól jön, de azt nem értem, hogy miért kell az OCR feladatok elvégzéséhez az oldalböngésző mintaképeihez textúrás háttér. Ez csupán dísz, a méretet viszont növeli.

— Egy portugál nyelvű, 3 oldalas folyó szöveget kollégám hagyományos asztali faxgépről küldött (standard 100 dpi-s minőségben), én pedig faxmodemmel fogadtam. Ez felismerés közben hibaüzenettel megállította a Recognitát. A nyelv kiválasztása mellett megadtam a rendszernek, hogy a dokumentum faxon érkezett, és a 6. felismerési szintet választottam, az anyagot pedig oldalakra bontva tápláltam be. A megállás a felismerés vége felé (97%) történt. (Hiba a DTK működése során 0xCD.) Ha a felismerési szintet eggyel lejjebb vettem, vagy a faxolt minőség jelzőjét kapcsoltam ki, a hiba megszünt, de ilyenkor a beolvasás használhatatlan lett. Jobb minőségű bemeneti anyagokkal, ugyanezen beállításokkal a hiba nem lépett fel, és az eredmény is használható volt.

— Egy táblázatos szerkezetű naptár forráskép-zónáinak teljes kialakítását manuálisan akartam befejezni, ám meglepetéssel tapasztaltam, hogy a program nem enged táblacellát kialakítani néhány olyan helyre, ahol pedig szükség lett volna rá.

— Próbára tettem a rendszert azzal is, hogy egy WinWord képernyőképről fel akartam ismertetni a szöveget (12-es betűméret, 72 dpi), ám az eredmény használhatatlannak bizonyult. (Sajnos a rendszer csak a 200—400 dpi felbontási tartományban tud korrekt eredményt szolgáltatni.)

Cseppentő Árpád
csepi@drotposta.hu



Fordulat a vírusfronton I.

Célpont a 32 bites Windows

A számítógépes vírusok kutatása gyökeresen átalakult, amikor a Windows 95 megjelent a piacon, ugyanis a DOS alapú korábbi vírusok és a bootvírusok jelentős hányada nem volt kompatibilis az új operációs rendszerrel. Különösen az olyan trükkös vírusok szaporodása lassult le, amelyek a DOS nem dokumentált rendszerhívásait használták, vagy amelyek lopakodó technikát alkalmaztak. A vírusírók hamar felismerték azonban az új kihívást, és elkezdtek olyan vírusokat írni, amelyek már működtek Windows 95 rendszeren, később pedig célba vették a többi 32 bites Windows platformot is.

Mindenki hajlamos arra, hogy a könnyebb megoldást keresse. A vírusírók is ilyenek. Legtöbben nem rendelkeztek megfelelő ismeretekkel a Windows 95 rendszerszintű programozásáról, viszont hamar rájöttek, hogy a makróvírusok potenciálisan alkalmazhatók az új környezetben is, hiszen ezek gyakorlatilag nem függnek az adott operációs rendszertől és az alkalmazott hardvertől. Elég sokan írtak makróvírusokat, és mára már több ezer variánst tartanak nyilván a víruskeresők fejlesztésével foglalkozó cégek.

A tehetségesebb vírusírók beérték egy-két makróvírus megírásával, azután gyorsan beleuntak a Word Basic nyelvbe, az interpretált környezet korlátaiba. Azt gondolhatnánk, hogy szerencsére, de inkább azt mondhatjuk, hogy sajnos. Amikor ugyanis profilt váltottak, olyan vírusokat kezdtek írni, amelyek végrehajtható programokat támadnak meg, azokhoz kapcsolják magukat.

A 32 bites Windowsra írt vírusok száma 2000 februárjában 400 körül mozgott, egy évvel korábban még csak 100 ilyen típusú vírust ismertünk. Többségük nem is okozott nagy gondot, általában csak helyi fertőzésekről lehetett hallani. Másik változás, hogy egy éve ezeknek a vírusoknak a 20%-a működött Windows NT rendszereken is, túlnyomó többségük a legtöbb a Windows 95-re korlátozódott. Mostanában az összefoglalóan Win32-eseknek elnevezett vírusoknak közel a fele Windows NT alatt is működik, és sokan kompatibilisak a Windows 2000 rendszerrel is.

Az elmúlt hónapban nagy riadalmat keltett a W2K.Installer megjelenése. Ez a vírus csak Windows 2000 alatt terjed, de sokan felreértették ezt az információt, és úgy gondolták, hogy a W2K.Installer az egyetlen ismert vírus, amely képes Windows 2000-en is terjedni. Az igazság az, hogy ez olyan Win32-es vírus, amely „önkorlátozást" gyakorol. Lekérdezi az operációs rendszer verziószámát. Ha ez a szám 5, akkor működésbe lép, feltételezvén, hogy Windows 2000 van a gépen, ugyanis a félrevezető propaganda ellenére a Windows 2000 nem más, mint a Windows NT 5.0. Ha nem lenne ilyen verzió-ellenőrzés a vírusban, akkor Windows NT 4 és Windows 98 alatt is működne. A W2K.Installeren kívül viszont majdnem minden Win32-es vírus képes Windows 2000-en is terjedni. A Windows 2000-ben jó néhány olyan új szolgáltatás jelent meg (például az SFC — System File Checker), amely lassítja vagy észrevehetőbbé teszi a legtöbb Win32-es vírus terjedését, de alapjában véve nem emel eléjük áthatolhatatlan gátat.

Kezdeti évek

A Windows 95-höz írt első ismert vírus a W95.Boza volt, amely az oprendszerrel egyidejűleg jelent meg. A vírust egy Windows 95 béta-verzión fejlesztette ki az akkor még aktívan működő ausztrál VLAD vírusíró csoport két alapító tagja. Hosszú idő telt el, amíg más vírusírók is megértették a Windows 95 sajátosságait, és elkezdtek hasonló vírusokat írni. Egészen 1997-ig kellett várni a Windows 95 vírusainak tömeges felbukkanására, de ezek közül néhány már kiterjedt fertőzéseket is okozott.

1997 végére az első Win32-es (tehát Windows NT kompatibilis) vírus, a W32.Cabanas.A is megjelent. Ugyanaz a fiatal vírusíró alkotta, akinek a nevéhez a hírhedt WM/Cap.A makróvírus fűződik. A vírusíró „Jacky Qwerty"-nek hívta magát, és a 29A spanyol nyelvű vírusíró csoport tagja volt, nagy valószínűséggel Peruból.

A W32.Cabanas.A kompatibilis volt a Windows 95-tel, a Windows NT-vel és a Win32-essel is, de Windows 98-on és Windows 2000-en is terjedni tudna, annak ellenére, hogy ezek a rendszerek még béta-változatban sem voltak meg abban az időben, amikor a Cabanas készült. A Cabanas vírus egyszerűen rémálommá változtatja a Microsoft Win32-es kompatibilitási álmát.

Win32-es vírust sokkal komplikáltabb készíteni, mint makróvírust. A legtöbbjüket még ma is Assembly nyelven írják. Némelyikük 20, 30, vagy akár 100 KB-os Assembly kód. Ez hihetetlenül hangzik, különösen azoknak, akik még emlékeznek arra az időre, amikor egy 3 KB-os vírust igencsak méretesnek tartottunk.

Persze ha jobban belegondolunk, az egész DOS kernel és a végrehajtó környezet 100 KB körüli Assembly kód volt, és már abban az időben egy „nem dokumentált" operációs rendszerről beszéltünk. Ma egy 20 KB-os Assembly vírus senkit nem lep meg, nem is szólva a magas szintű programozási nyelvet használó borzalmakról, amelyek víruskódja néha 300-500 KB hosszú is lehet (például Delphiben készült Win32-es vírusok).

Mára már biztosra vehetjük, hogy a korai DOS-os vírusok kihalóban vannak, és helyüket az Win32-es új fájlvírusok veszik át. A folyamat a kezdeti szakasz végére ért, becslésünk szerint rövidesen több ezerre nő a 32 bites Windows vírusainak száma.

A Win32-es platform kiváló táptalajt jelent: minden Win32-es rendszer támogatja az ún. Portable Executable (PE) fájlformátumot. A Windows 9x, a Windows NT és az új Windows 2000 rendszer fő végrehajtható fájlformátuma a PE, amely nagyon emlékeztet a Unix COFF fájlformátumára, és ahhoz hasonlóan rendkívül jól hordozható, sokféle hardverkörnyezetben alkalmazható. Az említett operációs rendszereken a legtöbb EXE, SCR, DLL, OCX stb. fájl PE formátumú. Ugyanezt a formátumot használják a főleg nem Intel alapú architektúrák Windows CE operációs rendszeren is.

A PE fájlformátum azonos, ezért valójában csak a processzor típusa korlátozza a fertőző vírusok terjedését. Mivel a Windows NT fejlesztése nem folytatódik Alpha alapú rendszereken, joggal vélhetjük, hogy az Intel alapú víruskódok eredményesen terjedhetnek az új, homogén környezetben. Ma még nem ismerünk ugyan egyetlen vírust sem, amely képes lenne Windows CE rendszeren terjedni, de ennek fő oka az, hogy a Windows CE látszólag csak RISC architektúrát használ. Ma még a legtöbb otthoni PC-n Windows 95 vagy Windows 98 fut. Ez a helyzet csak lassan változik meg, mert a Windows 2000-et főleg olyanok használják majd, akik szeretik a Windows NT-t, és elsősorban a munkájukhoz használják a számítógépet. Ebből az következik, hogy a vírusírók jelentős része Windows 95-re fogja megírni vírusait, de biztosra vehetjük, hogy sokan közülük már tanulják a Windows NT vagy a Windows 2000 alapjait.

W95.Anxiety

Az első széles körben elterjedt 32 bites Windows vírus a W95.Anxiety család volt. Minden jel arra utal, hogy ezt a vírust ugyanaz a vírusíró írta (VicodinES álnéven), aki jelen pillanatban az amerikai bíróság döntésére vár a hírhedt Melissa makróvírus megírása miatt. (A Melissa minden idők leggyorsabban elterjedt vírusa.)

Az Anxiety vírus a VMM-be ágyazza be magát. A vírus rámutat a Windows 95 egyik leggyengébb pontjára, hogy minden felhasználószintű program különösebb erőfeszítés nélkül módosíthatja a rendszermagot. Ez azért lehetséges, mert a Windows 95 nem védi a rendszerterületet a felülírástól. A Microsoft nem tudta kellően támogatni a régi Windows 3.x-re írt VxD-ket (Virtual Device Driver — virtuális eszközmeghajtók), és inkompatibilitási problémákra hivatkozva nem támogatja ezt a nagyon fontos rendszervédelmi tulajdonságot. Ez azután a Windows 95 igazi gyenge pontjává vált. Ha például egy hibás program (mondjuk a CALC.EXE) elköveti azt, hogy érvényes memóriaterületre ír a 0xC0000000 (3 GB) határ fölött, akkor a rendszer többnyire összeomlik.

Az Anxiety a fájlrendszerhez láncolja magát, így minden fájlnyitási műveletre röptében tud reagálni, azonnal képes megfertőzni minden PE formátumú megnyitott EXE fájlt. Az Anxiety vírus egyik fő trükkje az, amire szerzője rájött: nem szükséges egy VxD meghajtót telepíteni ahhoz, hogy a vírus VxD rendszerhívásokat tudjon használni. Ehhez elég annyi is, hogy a vírus bevésse magát a rendszerterületre, és ha ott vezérlést kap, a megfelelő VxD rendszerhívás használata már gyerekjáték. Mivel a Windows NT (és a későbbi Windows 2000) nem támogatja a VxD-ket, az Anxiety vírus ezen rendszerek alatt nem is tud terjedni.

A korai 32 bites vírusok gyengeségei

Az első működőképes vírusokat a már említett 29A csoport fejlesztette ki. Egyik vírusuk, a W95.Marburg azért vált hírhedtté, mert 2 hónap leforgása alatt közel 10 számítógépes magazin CD-ROM mellékletén is „forgalomba került". A vírus majdnem teljesen Windows NT-kompatibilis volt, de egy aprócska hiba miatt mégsem működött NT-n. Ez a hiba a legtöbb első generációs Windows 95 vírusban előfordult. A Windows NT betöltő modulja több dolgot ellenőriz a PE fájlokban, mielőtt elindítja azokat. A Windows 95 azonban tökéletlen PE EXE fájlokat is elindít, míg a Windows NT óvakodik az alapvetően hibásnak ítélt fájlstruktúráktól. Első ránézésre úgy tűnhet, hogy a Windows NT-be kiváló vírusvédelem van beépítve, de közelebbről nézve kiderül, hogy szó sincs erről. Amikor a W95.Boza vírus megjelent, sok víruskutató tesztelte, hogy Windows NT alatt is működik-e. Hamar kiderült, hogy a vírust még elindulása előtt lebuktatja a Windows NT, és az alábbi szövegű hibaüzenetet jeleníti meg:

C:\LOOK\boza.exe is not a valid Windows NT application.

Ez a hiba azért jelenik meg, mert a PE fájl fejlécében egy mező helytelenül van kiszámolva, és így a fájl fizikailag rövidebb, mint amekkorának lennie kellene.

Az eredeti Boza vírusnak sok egyéb problémája is volt. Legnagyobb hibája, hogy „bedrótozott" címeket használ az egyes Win32-es API-k (rendszerhívások) eléréséhez. Ezek a rendszerhívások az operációs rendszerhez tartozó könyvtárakban (tipikusan a USER32 .DLL-ben és a KERNEL32.DLL-ben) tárolódnak, és az ezeket felhasználni kívánó programok innen importálják az eljárásokat.

A GetCurrentDirectoryA( ) belépési címei két Windows 95 verzióban:

Belépési cím	Ordinal
A. 00007744	304 GetCurrentDirectoryA( ) (Win95 – Angol)
B. 0000774C	307 GetCurrentDirectoryA( ) (Win95 OSR2 – Magyar)

Az első használt címnek az aktuális könyvtárat lekérdező GetCurrentDirectoryA függvényre kellene mutatnia. Ez a függvény a KERNEL32.DLL-ben van elhelyezve, a programok onnan „importálják" azt. Minden API-nak van ún. exportcíme, amely a DLL-en belüli fizikai fájlpozíciót adja meg. Ez a cím az adott DLL „exporttáblájában" van. A cím természetesen az adott DLL tartalmától függ, így a magyar Windows 95 változatban lévő API cím egy kicsit eltér az angol verzióétól a magyar nyelv támogatása miatt. Ez az oka annak, hogy a Boza valójában sok más Windows 95 verzióval sem kompatibilis.

A W95.Marburg vírusban (és a W32.Cabanas.A-ben is) van egy nagyon fontos trükk, mellyel a mai vírusok szinte mindegyike él: képesek megállapítani az egyes operációs rendszereken az összes meghívni szándékolt API függvény címét.

Minden API cím az őt tartalmazó DLL-ből érhető el. A programozók egyszerűen az API függvény nevét használják, és a linker feladata, hogy a program importtáblájában a megfelelő API neve szerepeljen. Ez az ún. statikus import. Mivel néhány API eljárás nem minden Win32-es környezetben érhető el, vagy másutt eltérően működik, a programozók az ún. dinamikus importot is használhatják. Ehhez a GetProcAddress( ) API-t használják. A GetProcAddress( )-t a linker az adott program importtáblájába rakja, így ennek az API-nak a címe elérhető lesz a program számára.

Mivel a vírusokra jellemző, hogy nincs importtáblájuk, a vírusírók egy olyan vírusmodult fejlesztettek ki, amely képes a GetProcAddress( ) címét az összes létező Win32-es környezetben megállapítani. Ha a vírus tudja ezt a címet, azon keresztül már az összes API függvény címéhez megbízható módon hozzáférhet. A GetProcAddress( ) címének megállapítása pedig általában úgy történik, hogy a vírus a memóriában megkeresi a KERNEL32.DLL betöltési címét. Ha ez a cím rendelkezésre áll, akkor már gyerekjáték a DLL exporttábláját végigböngészni a memóriában, majd a GetProcAddress( ) címéhez hozzájutni.

Amikor egy 32 bites Windows vírus ezt a módszert használva a PE programot helyesen módosítja, akkor nagy valószínűséggel működni fog Windows NT vagy Windows 2000 alatt is.

Szőr Péter
pszor@symantec.com

(A második rész a következő számban.)

A partíciók varázslója

Hogyan szervezhetjük át lemezfelületünket?

Az [Új] Alaplap néhányszor foglalkozott már azzal, hogy több operációs rendszert is lehet telepíteni ugyanarra gépre. Ehhez elengedhetetlen a merevlemezek particionálása és a partíciók karbantartása. Egyes programok a komplett partíciók mentését és visszaállítását szolgálják (amilyen például a Norton Ghost, lásd Új Alaplap, 2000. január), mások pedig a lemezpartíciók átalakításában lehetnek segítségünkre. Ez utóbbiak közé tartozik a PowerQuest cég PartitionMagic új (5-ös) verziója, amely kezelni tudja a Windows 2000 fix partícióit is.

A ParticionMagic dokumentációja felhívja a figyelmet arra, hogy a program ugyan kezeli a Windows 2000 partícióját, de nem szabad arra telepíteni, hanem egy DOS által is kezelhető, hagyományos FAT16-os fájlrendszerű elsődleges partíciót kell neki kijelölni. Az automatikus telepítőprogram Windows 9x platformra készült, bár a készletben megtalálhatjuk azokat az állományokat is, amelyekkel 16 bites DOS/Windows 3.x felületen használhatjuk a programot, sőt Linux rendszerben készíthetjük el az indítófloppykat.

Első „primary", második „primary"...

Amikor Win9x alól indítjuk el a telepítőprogramot, célszerű a teljes telepítést választani. Felajánlja a DOS-os indítólemezek készítését, továbbá a BootMagic telepítését biztonsági floppyval együött. A BootMagic nem igényel túl sok helyet, és amíg nem telepítünk a gépre második, harmadik, x-edik rendszert, nem szükséges aktivizálni, észre sem vesszük jelenlétét. (A BootMagicre később még visszatérünk.)

A DOS-os verziót telepítés után a PartitionMagic \DOS könyvtárában fogjuk megtalálni. Ez azért fontos, mert ha olyan partíciót kívánunk módosítani, amelyen a Windows pótmemóriája (swap területe) foglal helyet, biztonságosabb DOS-ban dolgozni, és kényelmesebb is, ha az ehhez szükséges állományok rajta vannak a merevlemezen, mert akkor nem kell a floppyval piszmogni. Ha DOS alól látjuk a CD-olvasót, akkor a CD-n található állományok segítségével ugyancsak elindíthatjuk a partíciók átszervezését.

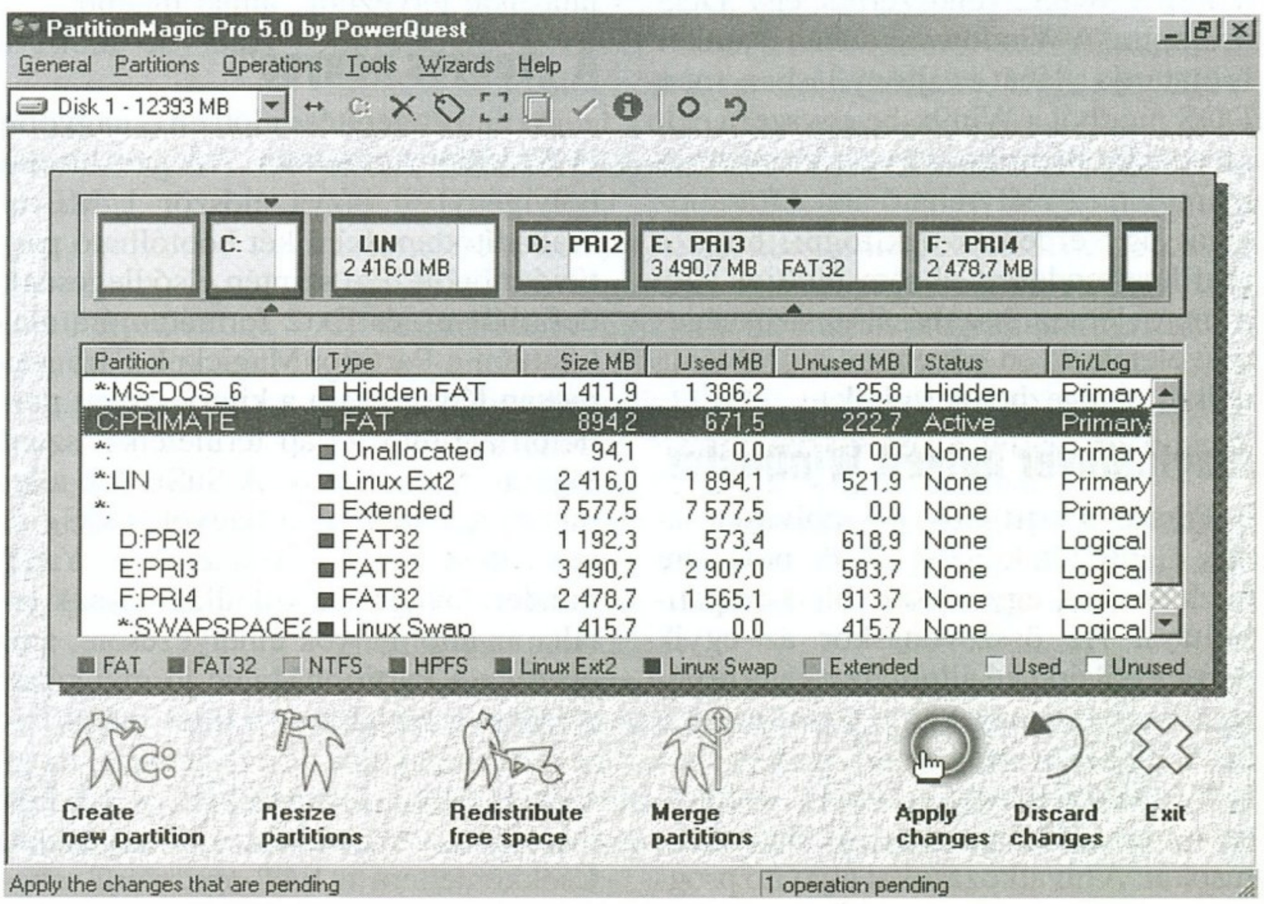
Ha előre tudjuk, hogy gépünkre több operációs rendszert akarunk telepíteni, és merevlemezünk mérete ezt lehetővé teszi, legjobb már a szűz lemezen létrehozni a megfelelő partíciókat. Már amennyiben ez sikerül... akadályok ugyanis vannak. Nekem például problémát okozott a 12 GB-os merevlemez birtokbavétele, mert (minő maradiság!) igényt tartottam a 16 bites DOS-ra is, amelynek FDISK programja nem kívánta látni a lemez túlnyomó részét. Miután szakmai tanácsokat kértem, létrehoztam egy 2 GB-os elsődleges partíciót (ekkorát volt hajlandó megalkotni), erre felkerült a DOS és egy Win95 OSR2, mely utóbbi FDISK-je már képes kiterjesztett partíciókat is létrehozni és kezelni, továbbá némi javítással a betöltési elágazás (dual boot) lehetőségét is megengedi.

Utána következhetett a többi operációs rendszer telepítése. A Win98-nak értelemszerűen saját elsődleges partíció dukál, és elsőként annak kialakítását bíztam rá a PartitionMagicre. Jól vizsgázott. Gond nélkül megoldotta, hogy az első primary, illetve a kiterjesztett partíció rovására egy második primary partíciót is kialakítson. (Groteszk ez a „második első", de ez van, ezt kell szeretni.) A telepítéshez a második primaryt kellett aktívvá tenni, a gépet floppyról indítani, és utána telepítni a Win98-at az új partícióba. Mivel ez egy önző rendszer, az eredeti állapotot a PartitionMagic indítólemezeinek felhasználásával lehetett helyreállítani, később pedig szükség volt a BootMagic aktivizálására is, mindkét rendszert (a Win95-öt és a Win98-at) fel kellett venni az indítómenübe.

FAT16 és FAT32

Dicséretes a program alkotóinak elővigyázatossága. Ugyanis a PartitionMagicet Win98 alá telepítve a BootMagic telepítéséhez érve intelligensen közölte, hogy már megtalálta magát egy példányban, és különben is csak az első primary partícióból életképes, tehát őt nyugodtan hagyjam ki a PartitionMagic telepítésekor.

Az pedig, hogy a fenti körülmények között a sokat emlegetett FAT16-os primary partícióból probléma nélkül használható volt, egyben azt is jelentet-



te, hogy nem szükséges a BootMagic számára külön DOS partíciót létrehozni, jól megfér a DOS/Win95 páros által birtokolt partíción. De követhetjük a hagyományos gyakorlatot is: 8-10 MB-os DOS partíció létrehozását, és egy 16 bites DOS telepítését. Ide a [cdrom]/ENGLISH/BTMAGIC/DOS/ könyvtárból telepíthető a BootMagic.

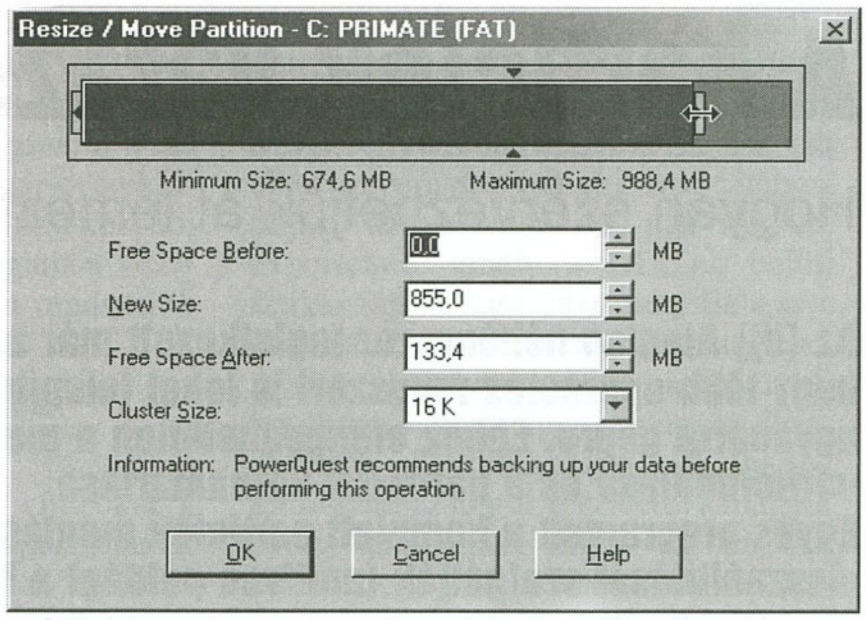
A csomag fő eszköze azonban mégiscsak a PartitionMagic, amellyel igen változatosan nyúzhatjuk merevlemezünket. Saját kísérleteimben szerepelt például a pótlólagos partíció létrehozása. A már említett, a Win98-nak szánt partíción kívül először kialakítottam egy plusz logikai meghajtót a kiterjesztett partíción. Csökkentettem a már meglevők terjedelmét, de még mindig maradt minden a DOS/Windows rendszerek formátuma körében. Kezdetben leragadtam a FAT16-os fájlrendszernél, majd elvégeztem a Win98-nak állítólag jobban tetsző FAT32-re történő konverziót, miáltal az is értelmét vesztette, hogy állandóan sok kis meghajtón keressem a dolgaimat. A nagy káosz szebb, és ennek szolgálatába szegődve nemcsak átkonvertáltattam a fájlrendszert, hanem össze is vontam néhány meghajtót. Kipróbáltam ugyanis a PartitionMagickel települő MagicMovert is, amely éppen arra készült, hogy kijavítsa a megváltozott útvonalneveket az állományokra hivatkozó programokban.

A fenti műveletek során tapasztaltam, hogy a PartitionMagic nem szereti az olyan helyzeteket, amikor swap állomány alól kell a partíciót konvertálni és átszabni — annak ellenére, hogy a legtöbb művelet elvégzéséhez kimenekül a Win32 rendszerből egy DOS promptra. A Windows azonban ilyenkor is ott tartja a lábát az ajtónyílásban, mert DOS módból a Win9x-be egyszerű exit utasítással bármikor visszatérhetünk. A merevlemez mélyreható átalakításához tehát úgy érdemes nekifogni, hogy a grafikus rendszert el sem indítjuk. Így a merevlemez átszabásához szükséges műveletek az én gépemen is probléma nélkül elvégezhetők voltak.

MagicMover helyett WinDoctor

Persze a partíciók összeolvasztásának alapfeltétele azok közös nevezőre hozhatósága, egymással való kompatibilitása. Az összevonáskor az egyik partícióról érkező állományokat a program beilleszti egy könyvtárba, ezzel a későbbiekben elvégzendő rendezéshez jó kiinduló feltételeket teremt, és kiküszöböli a névütközésből eredő konfliktusokat. A hivatkozásokat átjavító program alkalmazására csak az új könyvtárstruktúra kialakítását követően célszerű sort keríteni.

Ha a megváltozott útvonalak miatt a Win32 indulásánál gondok lennének — például egy automatikusan induló program miatt —, akkor segíthet az F8/"safe" indítás, majd át kell nézni ezeknek a programoknak az útvonalait, és szükség esetén néhány bejegyzést manuálisan kell átjavítani. Mi tagadás, nekem a MagicMover használatánál hatékonyabbnak tűnt a Norton Systemworks-ben található WinDoctor. Ez képes az összes változást automatikusan felderíteni, összegyűjteni, majd az automatikus javítás engedélyezése után a hibákat korrigálni.



Amíg a meghajtók betűjelét a változtatások nem érintik, addig kevés előkészülettel és egy kis utómunkával jól elboldogulunk a PartitionMagic használatakor. Előkészületként érdemes a Scandisk és a Defrag elindításával rendet teremteni a merevlemezen. A PartitionMagic működés közben már önmaga gondoskodik a folyamatos ellenőrzésről, kezdve a hibás szektorok felderítésével az állományok folyamatos ellenőrzésén át a végső integritásvizsgálatig. A műveletek elvégzése előtt ajánlatos háttérmentést készíteni az egész merevlemezről, de a különösen fontos (nem pótolható) állományokról mindenképpen. Minél mélyebb változtatásokat tervezünk, annál inkább.

A „vegyes négyes"

A Linux telepítése előtt én is nagyobb előkészületeket tettem. Az ígért kisebb helyigényben bízva először 1 MB-ot szabadítottam fel a két bootolható partíció mögött. Ezt szintén elsődlegesnek definiáltam, és Ext2 formátumúra alakítattam a PartitionMagickel. Párhuzamosan létrehoztam a kiterjeszetett partíción a Linux swap területének szánt logikai meghajtót is. A SuSE 6.2 telepítését a Win95 alatt hagyott DOS-ból indítottam el. A területeket a YaST minden további nélkül alkalmasnak találta az állományok elhelyezésére, ami igazolta a PartitionMagic jó munkáját. Később a kialakított terület mégis kicsinek bizonyult a StarOffice-szal megspékelt Linux számára, és ez további merevlemez-átalakítást vont maga után. Csökkentettem a FAT rendszerű meghajtók, majd a kiterjesztett partíciók méretét, és további 1 MB-tal növeltem a Linux Ext2-es területét. A Linuxban készített DOS floppykkal elindított PartitionMagic az elllenőrzéseket is beleértve probléma nélkül birkózott meg a feladattal.

A Linux telepítését követően a Lilo működését saját partíciójára korlátoztam, és felvettem a Lilót a BootMagic menüjébe. A későbbiekben ennek indítása a PartitionMagic-kel telepített bootmenedzserből sem okozott problémát. Tekintve, hogy a BootMagic a dokumentáció szerint is támogatja a Windows 95/98, Windows NT 4.0, Windows NT 3.51, Widows 3.x, MS-DOS v5.0 (6.22), PC-DOS 6.1 és későbbi PC-DOS-ok, Open DOS, OS/2 3.0 (4.x), Linux, BeOS, valamint a legtöbb MS-DOS klón alkalmazását, tapasztalataim szó szerint azt mutatták, ami a „nagy könyvben meg van írva".

Az általam vizsgált esetekben a csomag alapprogramja, a PartitionMagic is jól vizsgázott a DOS v6.22/Win95 /Win98/SuSE v6.2 alkotta vegyes négyes kialakítása és kezelése során. Az egyetlen problémát a Norton SystemWorks okozta, amikor alkalmaztam a szemétláda külön védelmét (Norton Protected Recycle Bin), és Win9x-ből indítottam a partíciókezelő műveleteket. Ilyenkor előfordult, hogy a védett állományok csoportját elveszett szektoroknak (lost clusters) érzékelte, és a változtatásokat felfüggesztve leállt. Ez azonban kiküszöbölhető a DOS-os indítással, tehát nem okoz komoly gondot.

Mérlegelve, hogy munkaráfordításban mekkora megtakarítást jelent, ha nem kell az egész merevlemezt „leradírozni", majd újra telepíteni ahhoz, hogy elvégezzük a partíciók átalakítását, szerintem mindenképpen érdemes a PartitionMagic használatára berendezkedni.

Simay Endre István

Java tanfolyam — 4. rész

Kommentárok, beágyazott osztály, névtúlterhelés, csomagok, JAR

A Java programnyelvről szóló sorozatunk negyedik részében a forráskódban elhelyezhető kommentárok típusairól, a névtúlterhelésről, a beágyazott osztályokról, a csomagokról és a JAR archív formátumról lesz szó.

A kommentárok

A Java forrásfájlokba a jobb érthetőség kedvéért megjegyzéseket helyezhetünk el. Ezekben megmagyarázhatjuk az egyes osztályok, változók vagy függvények szerepét és felhasználási módjait. Az ilyen kommentárok alapvetően két típusba sorolhatók: programozással kapcsolatosak, illetve dokumentációs kommentárok. A fordító mindent figyelmen kívül hagy, ami az ilyen kommentárokban található, azok emberi fogyasztásra készülnek. Lényegében nem képezik a forráskód részét, amit az is mutat, hogy minden következmény nélkül elhagyhatók. (Az alábbi ismertetés a Sun Microsystems Inc. ajánlásait tükrözi, más cégek Java fejlesztőeszközei esetleg más kommentártípusokat is engedhetnek, vagy az itteniekét mellőzhetik.)

I. A programozással kapcsolatos kommentárok

Ezek a forráskóddal vagy az adott megvalósítással kapcsolatos megjegyzéseinket tartalmazzák.

1. Fájlok, metódusok, adatok és függvények magyarázata

```
/*
     * szöveg
     */
```

Ezt a kommentárfajtát a fájlok elején, a metódusok előtt vagy a metódusokon belül használjuk, ha azt szükségesnek érezzük. Az ilyen kommentárokat egy üres sorral kell elválasztani a forráskód többi részétől.

2. Egysoros magyarázatok

```
/* szöveg */
```

Rövid megjegyzések esetén használatos. Előtte egy üres sort kell kihagyni:

```
     public Application01b() {
          /* ide jön a konstruktor törzse */
          ...
     }
```

3. Nagyon rövid kommentárok

A forráskód soraival egy vonalban is elhelyezhetünk egysoros magyarázatot, de ha több ilyen megjegyzést teszünk, akkor egy vonalba kell hozni őket:

```
class Application00a {
     public static void main(String[] args) {
                              /* A main metódus */
          System.out.println("Ratiosoft");
                              /* Ratiosoft kiíratása */
     }
}
```

4. Dőlt zárójeles határoló

```
// szöveg
```

A fordító figyelmen kívül hagy mindent, ami a // jelek után következik, egészen a sor végéig. Többsoros kommentár esetében csak akkor használjuk, ha el akarunk rejteni valamit a fordító elől, de valamilyen okból egyelőre nem akarjuk törölni azt a kódrészletet:

```
public class Application01 {
     // public Application01() {
          //
          //        ...
          //
          // }
     }
```

II. Dokumentációs kommentárok

A dokumentációs kommentárokat a JSDK szerves részét képező javadoc program használja, automatikusan kigyűjtve az általunk megadott forrásfájlokban lévő olyan kommentárokat, amelyeket ilyen határolók közé tettünk:

```
/**
 * dokumentáció
 */
```

A fordító figyelmen kívül hagyja a /** és a */ jelek közötti szöveget. Fontos, hogy dokumentumkommentárokat ne tegyünk metódusok és konstruktorok belsejébe, mert az ilyen megjegyzéseket a javadoc program mindig a megjegyzés után következő metódus elé teszi.

A dokumentációs kommentárok olyan programozók számára készülnek, akik végül a programnak csak a bináris kódjához fognak hozzájutni. Ez történik például az olyan feketedoboz technológiák esetében, amilyen a Windows, ahol sötét homály fedi az operációs rendszer működését. Ugyanakkor ezekről a féltve őrzött technológiákról bizonyos szintig mégis fel kell lebbenteni a fátylat, mert egyébként a Windows alá programozó szoftvermérnökök képtelenek lennének hatékony windowsos programok fejlesztésre. Azt pedig mégsem várhatják el tőlük, hogy kényszerűségből visszafejtsék a forráskódot. Nekik találták ki az API dokumentációkat.

A Windows operációs rendszerrel ellentétben a Java nyílt forráskódú rendszer, a teljes forráskód hozzáférhető, és minden megkötés nélkül tanulmányozható, bizonyos kötöttségek betartásával akár változtathatunk is rajta. Ennek ellenére a Javasoft programozói lehetővé teszik API dokumentációk létrehozását is a javadoc eszközzel, amely felépítése és tömörsége miatt gyakran áttekinthetőbb, mint maga a teljes forráskód. Mi magunk is használhatjuk ezt az eszközt saját API készítésére. Ekkor eltitkolhatjuk programunk forráskódját, ugyanakkor a fenti kommentártípus használatával annyi információt adhatunk ki róla, amennyit jónak látunk. Dokumentációt a számos osztályt tartalmazó, kiterjedt alkalmazások esetén azonban mindig érdemes készíteni.

Feladat: Másoljuk át a JavaDocExample.java nevű fájlt egy tetszőleges mappába, mi magunk is lépjünk át oda, majd adjuk ki a következő parancsot:

```
javadoc JavaDocExample
```

Célszerű előtte mindent kitörölni az adott mappából, hogy lássuk, milyen HTML formátumú fájlok jönnek létre. Majd hasonlítsuk össze az eredményt a JavaDocExample.java forrásfájlban lévő kommentárokkal. Gondolkodjunk el azon, hogy mi került bele a javadoc fájlokba, és mi nem, és mi lehet ennek az oka!

```
/*
 * @(#)JavaDocExample.java 1.0  2000/01/23
 *
 * Copyright 2000 Ratiosoft. //Copyright információk
 *
 */
/**
 * Ez az alkalmazás a Ratiosoft szövegkonstansot
 * és négy számot ír ki a standard kimenetre.
 * A JavaDocExample() metódus az alkalmazás
 * konstruktora. A main metódus nélkül nem futtatható!
 *
 * @version 1.0  2000/23/01
 * @author Szaló István
 */
public class JavaDocExample {
    /**
     * Ez egy statikusnak és véglegesnek deklarált
     * String. Ebben tároljuk a Ratiosoft szót.
     */
    static final String ratiosoft = "Ratiosoft";
                                   //Ratiosoft szövegkonstans
    /* int típusú egyedváltozó */
    public int i = 1;
    /** Egy double típusú változó, amit a number()
függvény használ. */
    protected double number = -1;
    /*
     * Egy double típusú változó
     * kezdeti értékadás nélkül.
     */
    private double n;
    /** Ez a konstruktor kezdeti értéket ad
    * az n változónak.
    */
    public JavaDocExample() {     // konstruktor
        /* Az n változó itt kap kezdeti értéket */
        n = -2;
    }
    /**
     * A main metódus a program belépési pontja.
     * Legalább egy osztálynak tartalmaznia
     * kell a következő lenyomatú metódust:
     * <p>
     * <code>public static void main(String
s[])</code>
     *
     */
    public static void main(String s[]) {
        System.out.println(ratiosoft);
      /* static, tehát nyomtatható! */
        //System.out.println(i);
      /* Az egyedváltozó nem nyomtatható! */
        JavaDocExample app = new JavaDocExample();
        System.out.println(app.i);
        System.out.println(app.number());
        System.out.println(app.number);
        System.out.println(app.number(app.n));
    } // end of main
    /**
     * A <code>number</code> változó neve és
     * a <code>number()</code> azonos,
     * de a () zárójelek megkülönböztetik őket!
     */
    public double number() {
        return number;
    /* A függvény egy double értékkel tér vissza */
    } // end of number
    /*
     * A fentivel azonos nevű függvény,
     * de ennek van paramétere,
     * így a fordító különbözőnek tekinti őket.
     */
    public double number(double _number) {
        return _number;
    /* A függvény egy double értékkel tér vissza */
    } // end of number
}
```

Figyeljük meg, hogy a javadoc kommentárok HTML címkéket (tageket) is tartalmazhatnak, ezek szintén beépülnek a dokumentációs program által létrehozott fájlokba. A HTML oldalak létrehozásához használatos nyelv szabályairól mindenki tájékozódhat a megfelelő szakkönyvekből. Megjegyzem még, hogy a jobb olvashatóság kedvéért a forráskódot célszerű tagolni: 4 szóközt használunk behúzásként vagy tabulátor esetében 8 szóközt, és kerüljük a 80 karakternél hosszabb sorokat, mert azok a szövegszerkesztőkben nehezen áttekinthetők.

Névtúlterhelés

A Java programnyelvben egyszerre több metódusnak is lehet azonos neve, ezt nevezzük névtúlterhelésnek. A fordító megnézi a paraméterek számát és típusát, eszerint tesz különbséget az azonos nevű metódusok között. Ebből következik, hogy nem deklarálhatunk azonos nevű metódusokat azonos számú és típusú argumentumokkal, legalább egynek ezek közül különböznie kell!

Feladat: Fordítsuk le az Application04 programot! Az alkalmazás most nem fogad el argumentumot a parancssorról, hanem egyszerűen kiírja az alpha és beta egyedváltozók értékét.

Az Application04 osztály

```
public class Application04 {
    public double alpha, beta;
    public Application04() {
        this(90,180);
    }
    public Application04(double alpha, double beta) {
        this.alpha = alpha;
        this.beta  = beta;
    }
    public static void main(String[] args) {
        Application04 app = new Application04();
        System.out.println("alpha = "+app.alpha);
        System.out.println("beta = "+app.beta);
    }
}
```

Az app objektum létrehozásakor a paraméter nélküli Application04( ) konstruktort hívjuk meg, amely a this( ) speciális metódussal meghívja a konstruktor paraméterezett alakját. Figyeljük meg a this kulcsszó kétféle használatát! A this( ) az osztály azonos nevű konstruktorait hívja, míg a this magára az osztályra hivatkozik.

Figyeljük meg, hogy a fenti JavaDocExample.java példában a number szó egy változó és két függvény neve!

Beágyazott osztály

Egy Java forrásfájl általában egy nyilvános elérésű osztályt tartalmaz. Ennek ellenére egynél több osztályt is deklarálha-

tunk benne, ahogy azt az Application04a.java fájlban láthatjuk. A fordítás után két .class kiterjesztésű fájlt kapunk:

```
1. Application04a.class
2. Degrees.class
```

Ez a gyakorlat mégsem ajánlott, célszerű mindig külön fájlba tenni az egyes osztályokat. Ha viszont valamilyen logikai vagy funkcionális szempontból fontos a két osztály egybetartozásának hangsúlyozása, akkor célszerű beágyazott osztályokat használni. Ekkor az egyik osztály beágyazódik a másikba:

```
class A {           // Beágyazó osztály
    ...
    class B {       // Beágyazott osztály
        ...
    }
}
```

Mivel a beágyazott osztály a beágyazó osztály belsejébe kerül, annak teljes körű hozzáférése lesz a beágyazó osztály minden tagjához, még akkor is, ha azok privát elérésűek. A beágyazott osztályok is lehetnek statikusak, ilyenkor statikus beágyazott osztályoknak nevezzük őket. Az ilyen statikus osztályok azonban nem hivatkozhatnak a beágyazó osztály egyedváltozóira és egyedmetódusaira, csak a már létrehozott objektumokon keresztül tehetik ezt meg (például: app.beta).

A nem statikus beágyazott osztályokat belső osztályoknak nevezzük. Ezeken belül nem deklarálhatunk statikus tagokat, ugyanakkor elérhetik az beágyazó osztály mindegyik egyedváltozóját és egyedmetódusát.

Az Application04b osztály

```
public class Application04b {
    final static double A_RADIAN = Math.PI/180;
    public Application04b() {
        this(0);
    }
    public Application04b(double alpha) {
        this.alpha = alpha;
        Degrees degrees = new Degrees();
        this.beta  = degrees.toRadians(alpha);
        gamma = degrees.degrees;
        delta = toDegrees(this.beta);
    }
    public double toDegrees(double anAngleInRadians) {
        return anAngleInRadians/A_RADIAN;
    }
    public static void main(String[] args) {
        Application04b app = new
        Application04b(Double.parseDouble(args[0]));
        System.out.println("alpha = "+app.alpha);
        System.out.println("beta
        (in radians) = "+app.beta);
        System.out.println("gamma
        (= beta) = "+app.gamma);
        System.out.println("delta
        (= beta in degrees) = "+app.delta);
    }
    class Degrees {
        double degrees;
        double toRadians(double anAngleInDegrees) {
            degrees = anAngleInDegrees;
            return anAngleInDegrees*A_RADIAN;
        }
    }
    private double alpha, beta, gamma, delta;
}
```

1. feladat: Fordítsuk le az Application04a és Application04b programokat! Mindkét program egy szám argumentumot fogad el a parancssorról, ismételten kiírja azt az alpha változóban, majd radiánná alakítja a betában, és vissza a deltában. A gamma változó a beágyazott osztály tagváltozójában tárolt értéket adja vissza.

2. feladat: Az Application04a programban írjuk át a konstruktorban a this(0) metódust this(0,0) alakra, majd fordítsuk le a kódot! Elemezzük a hibaüzenetet!

3. feladat: Az Application04a.java fájlban lévő Degrees.class osztályban lévő degrees tagváltozót tegyük privát elérésűvé (private double degrees;), majd elemezzük a hibaüzenetet!

4.feladat: Bogarásszuk ki az Application04c.java fájlt úgy, hogy a Degrees osztály végig statikus maradjon!

A továbbiakban figyeljük meg a következőket:

— Az Application04b program azáltal is hangsúlyozza összetartozását belső osztályával, hogy az most nem az önálló Degrees.class nevet kapta, hanem Application04b$Degrees.class lett. Ez a gyakorlat lehetővé teszi, hogy a fordítás után is tudjuk, mely osztályokhoz milyen beágyazott osztályok tartoznak.

— A privát elérésűnek deklarált változókat a program végére írhatjuk. A forráskódot olvasó programozót feltehetően csak a nyilvánosnak deklarált változók fogják érdekelni, hiszen azokat lehet módosítani. Ebből a megfontolásból kiindulva a privát tagok leghátulra kerülhetnek.

— A beágyazott osztályt is létre kell hozni, nem keletkezik automatikusan a beágyazó osztály létrehozásakor:

```
Degrees degrees = new Degrees();
```

— Az aRadian statikus változó privátnak van deklarálva, mégis elérhető a belső osztály számára.

— A belső osztály egyedváltozója az objektumhivatkozás szabályai szerint érhető el:

```
gamma = degrees.degrees;
          ⇑        ⇑
      Objektum Tagváltozó
```

— A konstruktor elfogadta paraméterként a parseDouble( ) függvény visszatérő értékét, azt nem kellett külön változóban tárolni:

```
Application04b(Double.parseDouble(args[0]))
```

— Felhasználtuk a Math osztály előre definiált PI értékét, amely a következőképpen van deklarálva:

```
public static final double PI = 3.14159265358979323846;
```

Csomagok

A valamilyen szempontból összetartozó osztályokat célszerű egyetlen csomagban összefogni. A csomagok használatával könnyebbé válik az egyes osztályok megtalálása, szabályozhatjuk az osztályokhoz való hozzáférést, és elkerülhetjük a névütközéseket. A csomag létrehozásához elegendő beírni a forrásfájlba a csomag nevét:

```
package ratiosoft.jardemo;
```

Ezzel létre is jött a csomag! Amikor egy program fejlesztését elkezdjük, akkor eleinte még nem rendezzük forrásállományainkat csomagokba, ilyenkor azok a név nélküli, alapértelmezett csomagba kerülnek. Ilyen névtelen csomagokat lehetőleg ne használjunk, csak kicsi, könnyen áttekinthető programok esetében!

A forrásfájl elején feltüntetett csomagdeklarációk egyben valódi könyvtárneveket reprezentálnak. A csomagnév minden egyes eleme megfelel egy létező alkönyvtárnak. Figyeljük meg, hogyan tükrözi vissza a java/code/package/ratiosoft/jardemo mappában lévő Degrees.class és Radians.class osztályok elhelyezkedése a bennük lévő csomagbejegyzést. Azt

is láthatjuk, hogy a Java nem abszolút címeket használ (például c:\java\code), hanem relatív címzést (például /ratiosoft/jardemo).

A kódkonvenció értelmében a csomagok neveit végig kisbetűvel írjuk, még akkor is, ha belső, értelmes tagokra lehet bontani őket! Nem ajánlott például a következő alak:

```
package ratiosoft.jarDemo;
```

A társaságok internetes tartománynevüket használják fordított sorrendben. Például a www.sun.com cím alapján képezték a következő csomagnevet:

```
package com.sun.tools.javadoc;
```

A com rész lehagyható:

```
package sun.io;
```

Nagy világcégek esetén a névütközések csökkenthetők, ha területi alapon is különbséget teszünk:

```
package com.sun.hu.packagename;
```

A javac fordítónak tudnia kell, hogy az egyes csomagok és az azokban található osztályok, változók és metódusok hol találhatók meg a könyvtárstruktúrában, és azok elérési helyét külön fel kell tüntetnünk a forráskódban. Erre három lehetőség van:

1. A csomagok osztályait egyenként importáljuk. A forráskód elején, közvetlenül a package bejegyzés után feltüntetjük az importálandó osztályokat (lásd a Jar01.java forrásfájlt a package2 mappában!).

```
import ratiosoft.jardemo.Degrees;
import ratiosoft.jardemo.Radians;
```

A csomagok neveit végig kisbetűvel írjuk, az osztályok nevei nagybetűsek.

2. Az egy csomagban lévő összes osztályt egyszerre is importálhatjuk:

```
import ratiosoft.jardemo.*;
```

3. Megtehetjük azt is, hogy nem használjuk az import kulcsszót, hanem az osztály minden egyes előfordulásakor a csomagnévvel együtt hivatkozunk rá (lásd a Jar02.java forrásfájlt a package2 mappában!):

```
ratiosoft.jardemo.Degrees degrees = new
ratiosoft.jardemo.Degrees();
```

Ugyanezt az írásmódot kell használnunk akkor is, ha osztályunkban egy olyan változónév vagy metódusnév van, amely ütközik az egyik importált névvel. Ilyenkor egyértelművé tehetjük, hogy helyi tagról van-e szó vagy importáltról.

A Java futtatókörnyezet a java.lang csomagban lévő osztályokat automatikusan behívja, eddigi programjainkba ezért nem kellett külön importálnunk a Double, Math és System osztályokat.

Feladat: Másoljuk át a package1 mappában lévő Jar01.java fájlt és a ratiosoft mappát egy ratio nevű mappába a merevlemezre, majd fordítsuk le a Jar01.java fájlt! Ügyeljünk rá, hogy a sikeres fordításhoz a Degrees.java és Radians.java fájloknak már lefordítva jelen kell lenniük a csomag által kijelölt mappában. A javac fordítónak most is meg kell adni az osztályelérési utat, de mivel a Degrees és Radians osztályok a ratiosoft.jardemo csomaghoz tartoznak, nem kell a teljes utat begépelni a –classpath opció után, ahogy eddig tettük, hanem azt a mappát kell az opció argumentumaként megadni, ahová a Jar01.java fájlt másoltuk, azaz példánkban:

```
Javac -classpath c:\ratio Jar01.java
```

Végezetül futtassuk le az alkalmazást a következő opciókkal:

```
1. java -cp c:\ratio\ratiosoft\jardemo Jar01
2. java -cp c:\ratio\ Jar01
3. java Jar01
```

Melyik esetben nem fut le az alkalmazás, és mi ennek az oka?

Feladat: Most töröljük le a Jar01.class fájlt, és írjuk be a Jar01.java fájl elejére a következő sort:

```
package ratiosoft.jardemo;
```

Majd fordítsuk le ismételten a programot, és futtassuk a fentiek szerint! Miért kapunk hibaüzenetet?

Feladat: Tanulmányozzuk a package2 mappában a Jar02.java fájlban lévő csomaghivatkozásokat, és hasonlítsuk össze azokat a Jar01.java fájlban lévőkkel!

A JAR archív formátum

Egy nagyobb program több tucat vagy akár több száz fájlból is állhat, s ezek egyenkénti letöltése a hálózatokon hosszabb időt vehet igénybe, mintha azokat egyetlen fájlba egyesítve töltenénk le. Ez volt az elsődleges szempont, amikor a Javasoft fejlesztői létrehozták a JAR archív formátumot. A JAR szó a Java ARchive szavak összevont rövidítése. Mivel elsősorban a platformfüggetlen csomagolás volt a lényeg, és a tömörítés csak másodlagos. Gyakran nem is tömörítünk a jar programmal, csak csomagolunk. A JAR fájlformátumot a ZIP fájlformátumra alapozták, ezért a ZIP tömörítők bizonyos fokig kezelni tudják a jar fájlokat, de nem teljesen. A jar tömörítő eszköz egy jegyzékfájlt is beletesz az archívba, és az alapértelmezésnek vagy a jegyzékfájl specifikációjának megfelelően mi magunk is írhatunk ilyen fájlt. A JAR tömörítés veszteségmentes.

A jar eszköz

A jar eszközzel csomagolhatjuk be és ki, tömöríthetjük vagy megtekinthetjük a JAR formátumú fájlokat. A DOS-os tömörítőkön „nevelkedetteknek" nem, de rögtön a Windows kényelméhez szokott felhasználóknak ez eleinte fapadosnak tűnhet. A parancssoron tudjuk elindítani, és több opciót is adhatunk neki:

–c Új és üres archívot készít a standard kimeneten (create).

–t A standard kimenetről kilistázza az archív tartalomjegyzékét (table of contents).

–x fájlnév A standard kimenetről kibontja a megnevezett fájlokat, vagy az összeset. Ha nem írunk be fájlnevet vagy neveket, akkor az összes fájlt kibontja az archívból, ha megnevezzük őket, akkor csak azokat (extract).

–f Az f opció megmondja a jar eszköznek, hogy nem a standard kimenetre kell írnia, vagy onnan olvasnia, hanem a megnevezett fájlból vagy fájlba (file).

–v Bő hibaüzenetet ad (verbose).

–m Előre megírt jegyzékfájlt szúr be az archívba (manifest).

–0 Nem történik ZIP tömörítés, csak becsomagolás (0 azaz zéró csomagolás).

–M Nem jön létre jegyzékfájl a becsomagolás során (Manifest).

–u Frissíti a már létező JAR formátumú fájlt. Új fájlokat szúr be, vagy a jegyzékfájlt frissíti (update).

–C A jar parancs végrehajtásakor könyvtárat vált (Change).

Feladat: Próbáljuk ki a következőket!

1. Írjuk be a jar parancsot a parancssoron! Az alkalmazás kiírja a fenti opciókat.

2. Most az iménti Ratio mappában adjuk ki a jar cf Jar01.jar * utasítást! Ez az adott könyvtárban lévő összes fájlt betömöríti a Jar01.jar fájlba. Így nemcsak becsomagolás, hanem ZIP tömörítés is történik. Megjegyzés: A WinZip 6.1-es verziója például nem tudja kicsomagolni az ilyen tömörített JAR fájlokat. Ismételjük meg a becsomagolást a 0 opcióval: jar cf0 Jar01.jar *, majd teszteljük újra a ZIP archiválót!

3. A jar xf Jar01.jar parancs kibontja a mappában lévő becsomagolt fájlokat. A Meta-inf mappában nézzük meg a

becsomagoláskor automatikusan létrehozott jegyzékfájlt (manifest.mf).

4. A jar cf Jar01.jar *.class parancs csak az adott mappában lévő class kiterjesztésű fájlokat csomagolja be.

5. A jar xf Jar01.jar Jar01.java parancs csak a megadott példafájlt csomagolja ki az adott mappába.

6. A jar xf Jar01.jar ratiosoft/jardemo/Degrees.class a megadott példafájlt bontja ki az alkönyvtárakkal együtt.

7. A jar tf Jar01.jar parancs kiírja a fájl tartalmát a standard kimenetre.

8. A jar tvf Jar01.jar parancs a v opció miatt részletesebb információt ad az archívban található fájlokról.

9. A jdk1.2.x mappában keressük meg az src.jar fájlt, amely a rendszer forráskódját tartalmazza. Ez nem azonos a teljes, újrafordítható forráskóddal, amely a Sun közösségi forráskód licenc keretében tölthető le a Javasoft szervereiről. Az src.jar fájl nem tömörített, és jegyzékfájlt sem tartalmaz. Miután egy tetszőleges mappába átmásoltuk, a jar xf src.jar utasítással bontsuk ki, majd keressük meg benne a Math.java osztályt! Szövegszerkesztővel nézzük meg, milyen matematikai függvények vannak benne, majd próbáljuk ezeket behelyettesíteni programjainkba! Például: anAngle = Math.abs(anAngle);

10. Ismerkedjünk a forráskóddal! Különösen azt figyeljük meg, hogy a csomagbejegyzések hogyan tükrözik vissza a tényleges könyvtárstruktúrát!

A jegyzékfájl

A jegyzékfájlba „meta" információkat helyezhetünk el becsomagolt állományainkról, ami új a ZIP formátumhoz képest. Ezeket a pluszadatokat sokféle célra fel lehet használni, és a platformfüggetlenség, a becsomagolás, valamint a tömörítés mellett ez adja a JAR formátum egyik nagy előnyét.

Az alapértelmezetten létrejövő META-INF könyvtárban egyetlen jegyzékfájl található, a MANIFEST.MF. A kis- és nagybetűs írásmódnak itt nincs jelentősége, de a nagybetűs írásmódot használjuk! Alapértelmezetten két bejegyzés kerül a jegyzékfájlba:

```
Manifest-Version: 1.0
Created-By: 1.2.1 (Sun Microsystems Inc.)
```

Ezeket a párokat egészíthetjük ki saját bejegyzéseinkkel. Például:

```
Required-Version: 1.0
Main-Class: Jar01
```

A jegyzékfájlban „név : érték" párok vannak elhelyezve. Az ilyen „név : érték" párok szekciókat alkothatnak, melyeket üres sorok választanak el egymástól.

Végrehajtható jar fájl

Ha becsomagolt állományaink futtatható programot alkotnak, akkor a jegyzékfájl Main-Class bejegyzése mondja meg a futtató környezetnek, hogy melyik .class kiterjesztésű fájl tartalmazza a main metódust. A végrehajtható jar fájlt a java -jar JarFile.jar paranccsal indíthatjuk el. A JarFile.jar helyébe kell beírnunk a futtatandó becsomagolt fájlokat tartalmazó archív nevét.

Kettőt kattintva a végrehajtható jar fájlokra, elindíthatjuk a programot, ha jól van beállítva a rendszer. A Windows 95 vagy 98 operációs rendszerben az Intéző nevű fájlkezelő Nézet – Mappa beállításai – Fájltípusok dialógusablakban ellenőrizhetjük, hogy hozzá van-e rendelve a jar kiterjesztés a javaw.exe programhoz (angol verzió esetén az Explorer fájlkezelőt keressük meg: View – Folder Options – File Types). Itt szerkeszthetjük az Executable Jar File bejegyzést, bár erre normális esetben nincs szükség. A javaw.exe program funkcióját tekintve azonos a java.exe programmal, de az utóbbi hívásakor egy parancssori ablak is megnyílik programunk fő ablaka mellett. Olyankor adjuk ki tehát a javaw parancsot, amikor a DOS ablakra nincs szüksége az általunk elindított alkalmazásnak.

Feladat: Egy szövegszerkesztővel hozzunk létre egy manifest.mft nevű szövegfájlt, majd másoljuk bele az alábbi sorokat:

```
Manifest-Version: 1.0
Main-Class: Jar01
Created-By: 1.2 (Sun Microsystems Inc.)
```

Úgy vettem észre, hogy ha a Main-Class bejegyzés nem a megfelelő helyen van a jegyzékfájlban, akkor a program egyszerűen nem indul el. Figyeljünk erre! A Main-Class bejegyzés után mindig annak a class fájlnak a nevét kell írnunk, amelyikben a programunkat elindító main( ) metódus található.

A mentés után tömörítsük be a mappában lévő fájlokat a következő paranccsal:

```
jar cmf manifest.mft JarDemo.jar Jar01.class
Jar01.java ratiosoft/jardemo ratiosoft/source
```

Most indítsuk el a létrejövő tömörített JarDemo.jar fájlt, a

```
java -jar JarDemo.jar
```

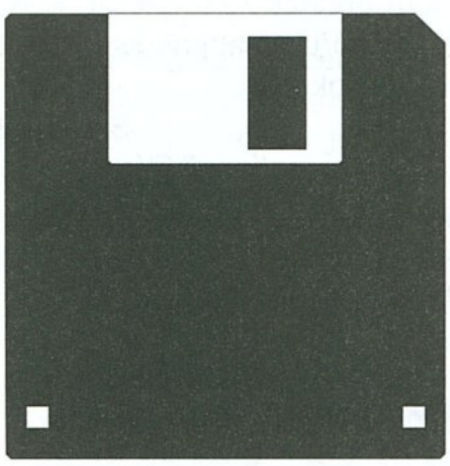
paranccsal a parancssoron! Útmutatásként a jardemo mappa tartalmazza a szükséges fájlokat, a jar mappa pedig a tömörített JarDemo.jar fájlt.

Bár a JarDemo.jar fájlra is kattinthatunk kettőt, nem sok minden történik, mert alapértelmezetten a javaw.exe indítja a .jar kiterjesztésű alkalmazásokat, a mi kis programunk viszont végzetesen rá van utalva a parancssorra. Hogy mégse fosszuk meg magunkat a kattintgatás élményétől, keressük meg a JSDK-ban a jdk1.2.x/demo/jfc mappát, és ott indítsuk el a .jar kiterjesztésű alkalmazásokat!

Szaló István
ratiosoft@freemail.c3.hu

E SZÁMUNK HIRDETŐI

SZOFTVEREK SOKSZOROSÍTÁSA FLOPPYRA, RÖVID HATÁRIDŐVEL



Részletes feltételek az Új Alaplap szerkesztőségében, Megyes Zsuzsánál, telefon: 322-4417.

Mikrobazár

A Mikrobazár rovatban a nem kereskedelmi célú egyéni hirdetések közlése ingyenes. A kereskedelmi célú apróhirdetések tarifája gépelt soronként (azaz 60 karakterenként) 300 forint. A terjedelem alapján így kiszámított összeget kérjük átutalni az Új Alaplap Kiadói Kft számlájára (OTP, 11706016–20788599), vagy feladni postai utalványon a kiadó címére (1539 Budapest, Pf. 571), és feltüntetni, hogy „Új Alaplap, apróhirdetés". A befizetést igazoló szelvény másolatát — a hirdetési szöveggel együtt — a szerkesztőséghez (a kiadóéval azonos címre) küldjék el.

Szerzői jogokat sértő szoftverhirdetéseket nem közlünk le.

Bármilyen típusú szöveg fordítását vállalom angolról magyarra, magyarról angol nyelvre, illetve vállalom kiadványok látványtervezését, szerkesztését is. Cím: Lachner Zoltán, 1195 Budapest XIX., Jáhn Ferenc u. 14/a. Telefon: 357-0308.

OBJECTS 2.0 — objektumorientált programozás CLIPPER-ben: www.tar.hu /proxima.

Adatmentés CD-re, streamerre; winchesterről, floppyról. Ugyanitt beszerzési tanácsadást, hálózattervezést és programkészítést is vállalok. Cím: Kovács Lajos, 1031 Budapest III., Vízimolnár u. 10. IV/33.

Alaplapcsere, memória-, winchester- és floppybővítés a helyszínen. MegaSoft. Telefon: 295-5085.

Stúdiómban megbízhatóan, ellenőrzötten lefordítom angol, német, francia és magyar nyelvről/nyelvre műszaki és közgazdasági folyóiratok cikkeit, hardver- és szoftverleírásait. Áfás számlát állítok ki. Cím: Szász György, 1035 Budapest III., Kórház u. 25. Tel.: 368-4874.

Súlyosan mozgáskorlátozott, bal kéz hiánnyal született gyermekem részére, akinek felnőtt korában egyetlen megélhetése a számítástechnikai ismeret lehet – **keresek olcsó** számítógép-kiegészítő hardver- (RAM, bővítőkártya, CPU stb.) és szoftverelemeket. Ajándékozás esetén adójóváírás megoldható. Cím: Szőke János, 5440 Kunszentmárton, Tanya 604. Telefon: (56)707-123.

Vásárolunk 5–7 db működőképes hercules (mono) monitort kártyával. Monitor max. 2500, kártya max. 600 Ft + áfa. Számla szükséges. Telefon: 347-7062, e-mail: jozsi@polifoam.hu

Keresem (esetleg kölcsönbe is) az **SSP, IBM System/360 Fortran Scientific Subroutine Package** leírását. Telefon: Szondi Egon János 463-1563. E-mail: szondi@reak .bme.hu.

Fényképek szkennelése, archiválása CD-n. SCI-FO Egyéni Cég. Tel.: 320-4702.

Programfejlesztő szakembereket keresünk ausztriai (bécsi) munkára

Követelmények:
- Internetes programozási technológiák ismerete.
- Java C/C++ programozási gyakorlat.
- Oracle adatbázis kezelése.
- Német, illetve angol nyelvtudás.

A szakmai önéletrajzokat a következő címre várjuk:

Austorex Kft

E-mail: radnoti@matavnet.hu

További érdeklődés: (30)952-6579

MÁJUSBAN A HÓNAP TÉMÁJA:

NETIKA

Számítógéppel emberi nyelven

Fokozatos közeledés a szemantikához

Az első problémám ezzel a könyvvel az, hogy vajon eljut-e mindenkihez, akit érdekelhet a téma. Lehet-e ma megfelelően „teríteni" ilyen kiadványt széles e hazában? Mert nagyon jó volna, ha sokan megismernék a könyvben felvetett problémákat. Nemcsak azok, akik egyetemen, főiskolán, tanfolyamok hivatalos kurzusain tanítják vagy hallgatják az intelligens szövegkezelés módszereit, hanem mindazok, akik elméleti vagy gyakorlati munkájuk során bárhol kapcsolatba kerülnek a számítógépes szövegkezeléssel.

Mindenki találkozhatott már szövegszerkesztőjében helyesírás-ellenőrző programmal, talán szidta is, amikor az olyasmit vélt hibának, ami nem volt az. Vagy (és ez a gyakoribb eset) nem figyelmeztette olyan hibára, amit észrevehetett volna, „ha van egy kis esze". Persze nincs, és ezért nem is lehet felróni neki, hogy nem vesz észre mindent. Inkább azon kellene csodálkoznunk, hogy ennek ellenére gyakran reagál úgy, mintha értelmes lény lenne. Sok mindenre meg lehet tanítani a gépet, ami eddigi fogalmaink szerint az emberi intelligencia priviléáiuma volt. Ilyen módszerek gyakorlati kidolgozásával (és elméleti kutatásával) foglalkozik az a kis csapat, amelyet Prószéky Gábor gyűjtött maga köré, és akik MorphoLogic néven tették magukat ismertté.

A magyar nyelv különösen gazdagon tárja a kutatók elé a megoldandó számítógépes nyelvészeti problémákat. (Csak úgy mellesleg: sok olyan nyelvész van a világban, aki kimondottan ennek a kedvéért tanult meg magyarul!) Mindenképpen érdemes odafigyelni, hogy miket mond tapasztalatai alapján Prószéky és mérnök-informatikus szerzőtársa az intelligens szövegkezelésről.

Nyelvészmérnökség

Négy nagy fejezetben foglalják össze mondanivalójukat: (1) A számítógépes szövegtárolás, ezen belül a kódolás, a rendezés, a formázás, a hipertext és általában a szövegszervezés. (2) Szöveg írása a számítógépen, gépi segítséggel. Ide értik a szerzők a helyesírás ellenőrzését, az elválasztást és a szinonimaszótárak problematikáját is. (3) A szövegben való keresés különböző módszerei. E fejezeten belül foglalkoznak a szerzők két testesebb témakörrel: a keresőgépekkel és a szótő-visszaállítás problémájával, de kitérnek a reguláris kifejezésekkel való keresés lehetőségeinek és korlátainak taglalására is. (4) A számítógépes fordítás és fordítástámogatás problematikája, a különböző fordítórendszerek működési elvének ismertetése.

Az egyes fejezetekhez kapcsolódik, de azoktól „nyelvtechnológiai alapműveletek" címen el is válik a módszertani kérdések részletesebb kifejtése. Olyan olvasók kíváncsiságának kielégítésére szolgálnak ezek a részek, akiknél a „mit" és a „miért" végiggondolása után már a „hogyan" kérdései is felmerülnek, és szeretnének bekukucskálni a gépi nyelvészettel foglalkozó szakemberek műhelytitkaiba. Egyúttal arra is választ kaphatnak a könyv érdeklődő olvasói, hogy milyen számítástechnikai eljárásokat fejlesztettek ki annak az új iparágnak a művelői, amelyet újabban „nyelvészmérnökség" (language engineering) néven szoktak emlegetni. Nyelvtechnológiai problémaként kezeli a könyv

(a) a morfológiai elemzést,
(b) a szótő-előállító program működését,
(c) a morfológiai generálást és
(d) a számítógépes mondatelemzést.

Érthető módon sokat hivatkoznak a szerzők saját „gyermekeikre", az egyre újabb változatokban megjelenő helyesírás-ellenőrzőre és a MoBiDic néven folyamatosan fejlődő számítógépes szótárrendszerre. A helyesírás-ellenőrzők még csak most kezdenek méltóvá válni nevükhöz, az eddigieket ugyanis szabatosabb lett volna inkább szóhelyesség-ellenőrzőknek hívni. Legfrissebb változataik már kezdik leküzdeni legfőbb korlátjukat: próbálják figyelembe venni a szó környezetét is. Tökéletesnek még nem tekinthetők, és az emberi intelligenciát nem is lehet teljesen kikapcsolni ezen a területen sem, de amit a MorphoLogic idáig elért, az tiszteletre méltó teljesítmény.

Sokan nem is tudják, hogy a magyar nyelvre kidolgozott morfológiai elemzőprogram több nyelvfüggetlen modult is tartalmaz. Egyes módszerei kis átalakítással jól alkalmazhatók például a német nyelvre, ahol, mint tudjuk, a hosszú összetett szavak miatt a szóhelyesség ellenőrzése és a szótározás szintén nem egyszerű. A MorphoLogic szótárrendszereinek is vannak világvi-



szonylatban figyelemre méltó tulajdonságai (például a több szótárban való egyidejű keresés imponáló sebességgel). Jó termékekkel a hátuk mögött a szerzőknek már van mire szerénynek lenniük, a külföldi partnerek egyre sűrűbben keresik velük az együttműködést.

Morfológiai elemzés

Szemléltetésként bemutatom, milyen problémákon kell átvergődnie egy magyar morfológiai elemzőnek, ha egy számára nem ismert szóról akarja megállapítani, hogy az milyen szótőnek melyik alakja, és hogy milyen szófajhoz tartozhat a kezelendő ismeretlen szótő. (A példa alig valamivel rövidebben megtalálható a könyv 131. oldalán.)

Legyen a keresett szóalak: kacsónak. A program először a változatlan alakot keresi, hátha alapszóként megtalálja. Sajnos nem — de elvi lehetőségként ez soha nem zárható ki.

Ekkor a végétől kezdve keresni kezdi a lehetséges szegmentálási határokat. Helyes lehet-e a KACSÓNA+K szegmentálás? A K végződés miatt lehetne többes számú főnév, akkor viszont a végső magánhangzó a-ról á-ra változna (vö. alma — almák).

Talán a KACSÓN+AK szegmentálás helyes lesz? Lehetne az -AK is a többes szám jelének egyik változata, ha létezne KACSÓN főnév. A programot azonban nem könnyű becsapni. Kivétel nélkül minden olyan főnevet ismer, amelynek többes száma -AK végződést kap, úgy, mint HÁZ+AK, VÁR+AK. Ezek a szavak ugyanis véges szótárral felsorolhatók a magyarban — a keresett szó azonban nem tartozik közéjük. (Az ilyen főnevek halmaza nem bővül tovább a mai magyar nyelvben. Nyelvész tolvajnyelven szólva: nem produktív a többes szám jelének -ak alakváltozata.)

Próbáljuk meg a KACSÓ+NAK szegmentálást. Ez jónak tűnik, de még nem egyértelmű. Ugyanúgy lehetne egy főnév részes esete, mint egy ige többes szám 3. személyű alakja. KACSÓ igét viszont nem ismer a rendszer. A magánhangzóra végződő igék száma egyébként is erősen korlátozott. Az igék halmaza ma is bővülhet ugyan, de csak a magyar nyelv szabályainak megfelelő módon, igeképzővel. Névszóból például ma is keletkeznek új igék, ezeknek a végén azonban mindig megtalálható az ún. denominális igeképző, amely a névszót (nomen) igévé változtatja. A KACSÓ alakban ilyen képzőt nem fedezhetünk fel.

Továbblépve: KACS+Ó+NAK. Ez talán már jó, hiszen egy „kacsni” ige melléknévi igeneve éppen ez lenne. Sajnos azonban megint csalódnunk kell: a produktivitás szűrője ezt is kiszűri. Fentebb már láttuk, hogy a mai magyar nyelvben nem keletkeznek újabb képzőtlen igék, a KACS ige pedig nem szerepel a gép szótárában. (Más szavakkal megfogalmazva: a képzőtlen igék halmaza is zárt, az sem bővül újabb igékkel.)

Menjünk tehát tovább. Legyen a szegmentálás KA+CSÓNAK. A „csónak” szót meg is találjuk mint önálló, ragozatlan főnevet, ebből pedig alkotható lenne összetett szó. Persze ha lenne ehhez egy KA főnév.

Keresési procedúránknak tehát három esetben lehetne pozitív eredménye, ha figyelembe vesszük a mai magyar nyelv viselkedésének szabályszerűségeit is: lehetne egy KACSÓNAK ragozatlan szóalak, aminek lehetséges szófajáról egyelőre semmit sem tudunk. Lehet egy KACSÓ főnév, ami pillanatnyilag nincs ugyan benne a szótárunkban, de benne lehetne. Végül lehetne összetett szó, amelynek CSÓNAK az utótagja, előtagja pedig valamilyen ismeretlen KA szó. Hipotetikus tőszóként felvehetjük akár mindhárom tövet a szótárba, és elkezdhetjük gyűjteni róluk az információkat. Tegyük fel, hogy később találkozunk egy KACSÓVAL alakkal. Ekkor már bátran beírhatjuk a KACSÓ szótő nyelvtani információi közé, hogy mély hangrendű. Megfelelő mennyiségű szöveg vizsgálata után bizonyára az is kiderül, hogy a másik két hipotetikus tőnek a valószínűsége a nullához tart.

Tanulórendszerek

A tanulórendszereknek különösen kétféle esetben van nagy jelentősége:

(1) Ha olyan rendszert fejleszt valaki, ahol roppant nagy munkát jelentene a nyelvtani információk összegyűjtése és a rendszerbe való betáplálása. Ezeket az információkat egyszerűbb lehet a szövegből kinyerni.

(2) Előfordul, hogy a szóanyag összegyűjtése maga is hatalmas feladatot ró a rendszerépítőre, például azért, mert az adott területen szinte követhetetlenül gyorsan nő a használt szókincs. Ilyenkor általában a szó környezetére vonatkozó információkat is érdemes gyűjteni. Pontos meghatározásra várnak ugyanis a nyelvtani jegyeken kívül a jelentésre vonatkozó információk is, ez pedig a környezet figyelembevétele nélkül szinte lehetetlen feladat. Ma már elmondhatjuk, hogy a számítógépes nyelvészetnek fejlett alágazata az ún. korpusznyelvészet, amely nagyméretű szövegek statisztikai vizsgálatából szerez megfelelő nyelvi ismereteket. A korpusznyelvészet ismert feladata angol szövegek esetében a szókörnyezet alapján a szavak szófajának meghatározása (part-of-speech-tagging, POS-tagging). Az ebből kapott eredmény egyúttal kitűnő nyersanyagot szolgáltat statisztikai alapú grammatikai vizsgálatokra és szintaktikai adatgyűjtés elvégzésére is. Mióta lehetővé vált hatalmas szöveganyagok gépi feldolgozása, a nagy szótárkészítő műhelyek rendszerint hatalmas annotált korpuszok alapján dolgoznak, vagyis a szövegekhez hozzáépítik a belőlük kipréselt információkat is. Az ilyen annotált korpuszok biztos támpontot adnak a szóanyag kiválogatásához éppúgy, mint a jelentések és a grammatikai segédinformációk előállításához.

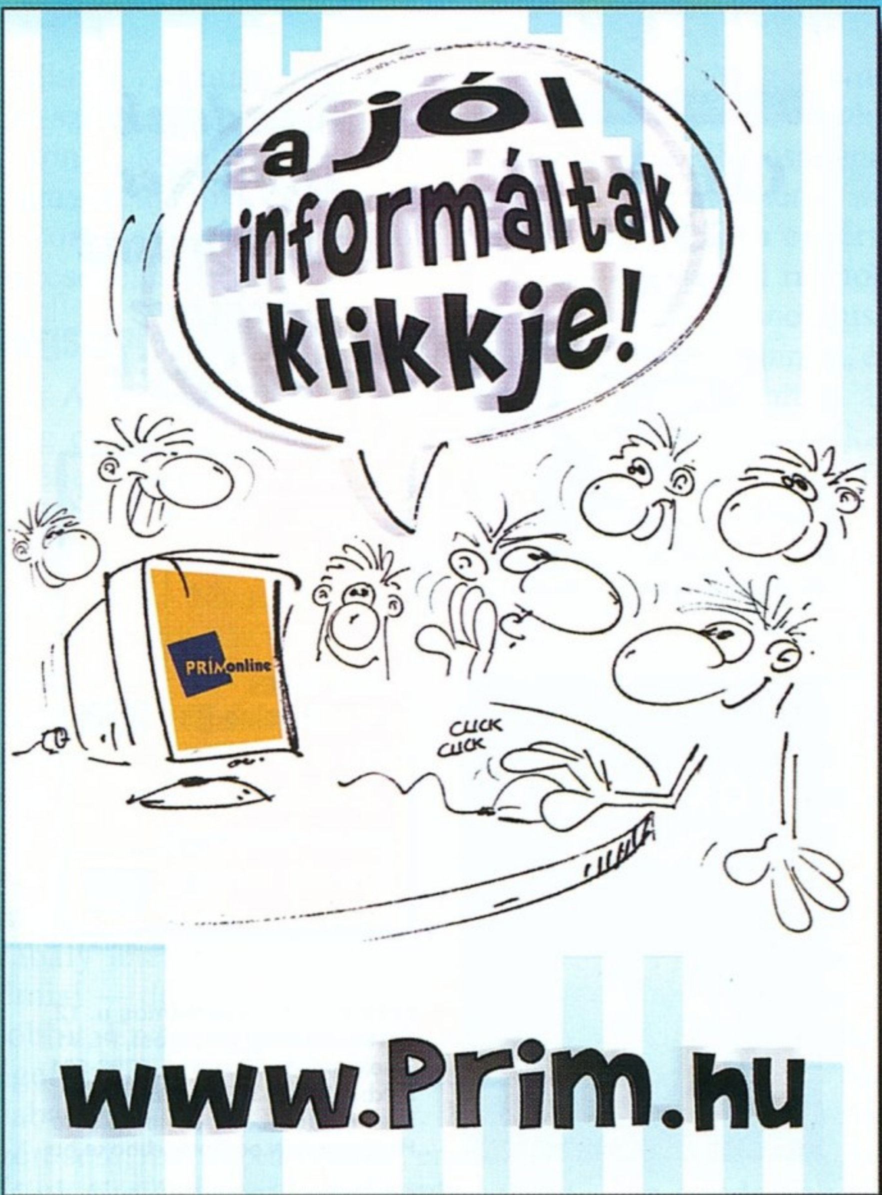
Rendkívül érdekes és jól kidolgozott része a könyvnek a fordítás folyamatának számítógéppel való segítése. A fordítási munkafolyamat elemzése során megismerkedhetünk mindazokkal a segédeszközökkel, amelyek az elmúlt évtizedek folyamán bebizonyították hasznosságukat. Bemutatják a szerzők az intelligens szótárkezelő rendszereket, a terminológiakezelő rendszert, a fordító keze alá dolgozó előfordító rendszert és az ún. fordítómemóriát. Hasonló számítógépes környezet építhető ki a lektor számára is. Bár a teljes folyamat egyes elemei még nem léteznek mindenki számára hozzáférhető formában, körülbelüli képünk alakulhat ki arról, hogy milyenek is lesznek a közeljövő fordítóműhelyei. Elsősorban ebből a fejezetből értheti meg az olvasó, hogy milyen jogon beszélnek a szerzők a számítógépes szótárak intelligenciájáról, és hogy milyen tevékenységekben jelenik meg ez az intelligencia.

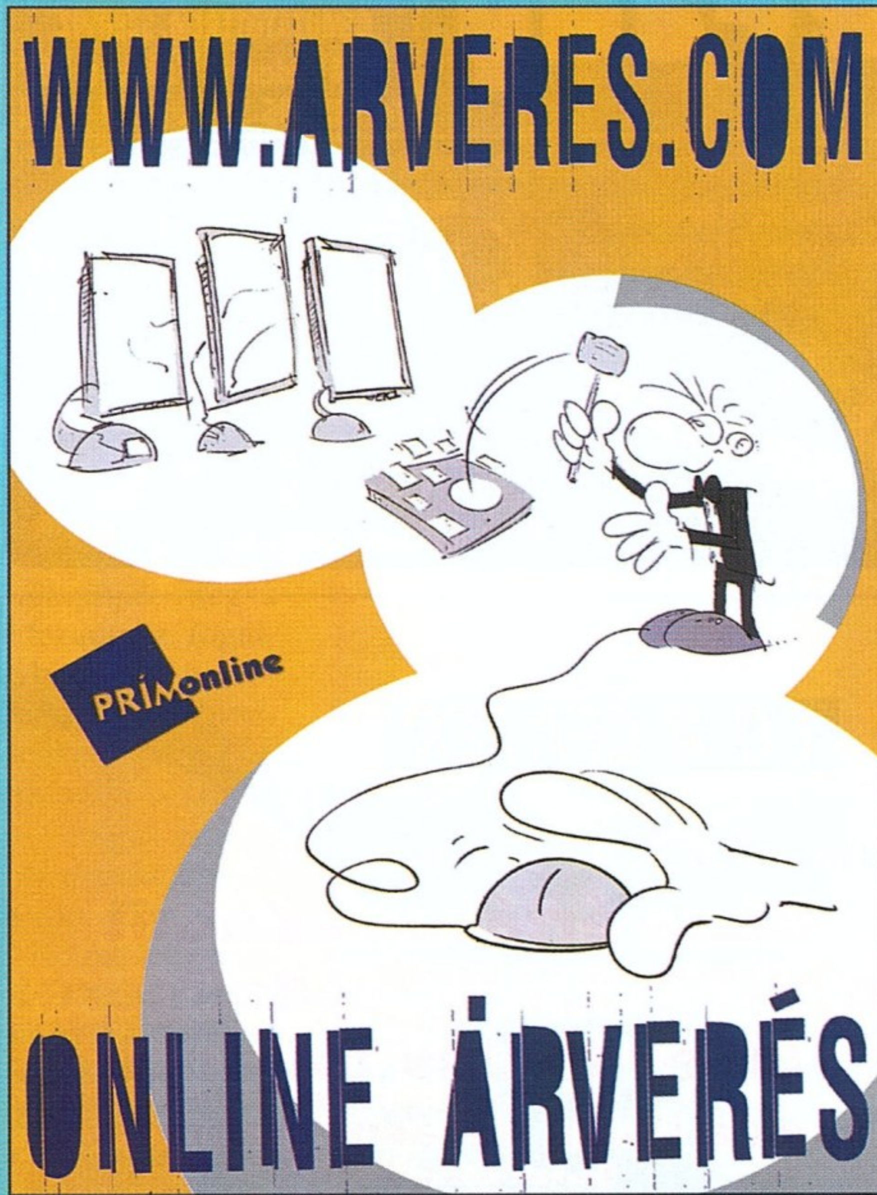
Nem állítom, hogy a könyv hibátlan. Helyenként nem elég pontos a megfogalmazás, másutt a szerzők is elmondják, hogy idő- és helyhiány miatt nem fejtettek ki olyasmiket, amiket nagy vonalakban már végiggondoltak. Amit azonban leírtak, az gondolatokban

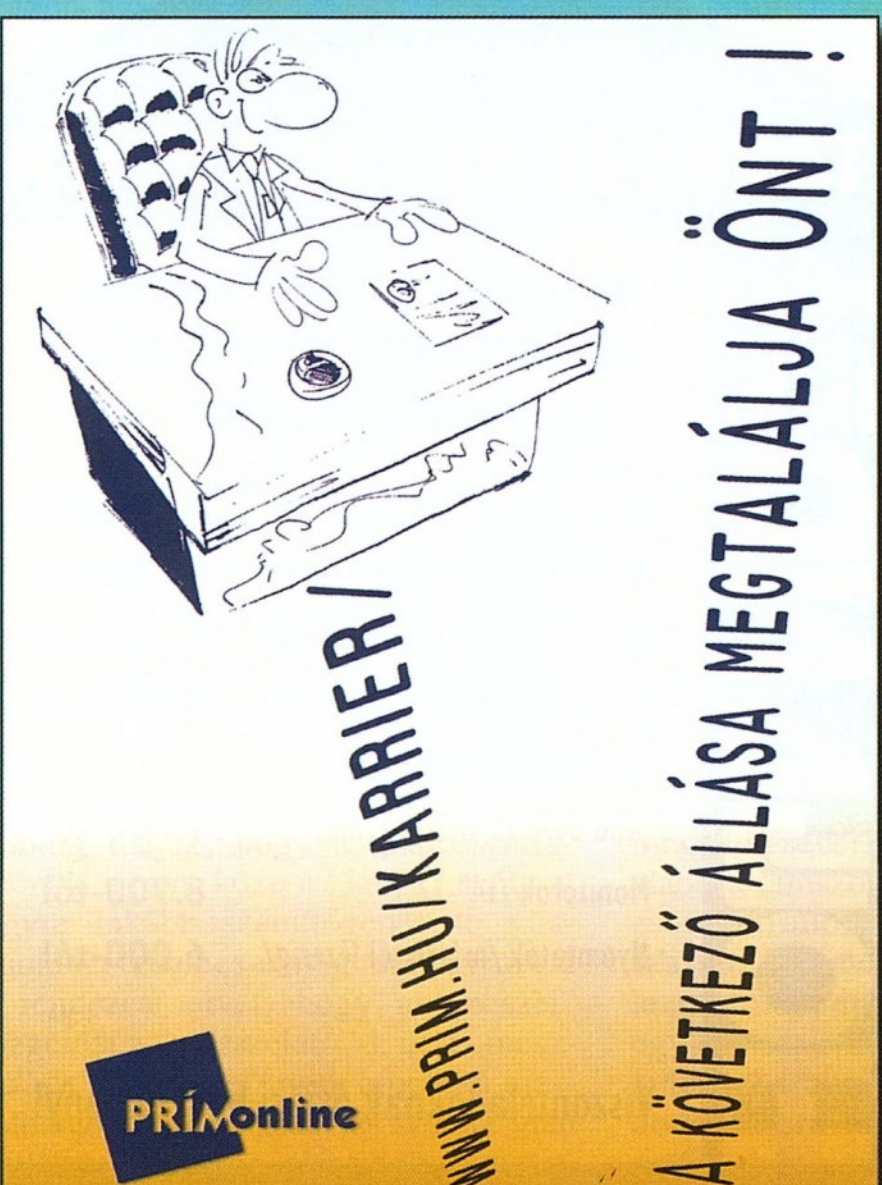
Prószéky Gábor — Kis Balázs:

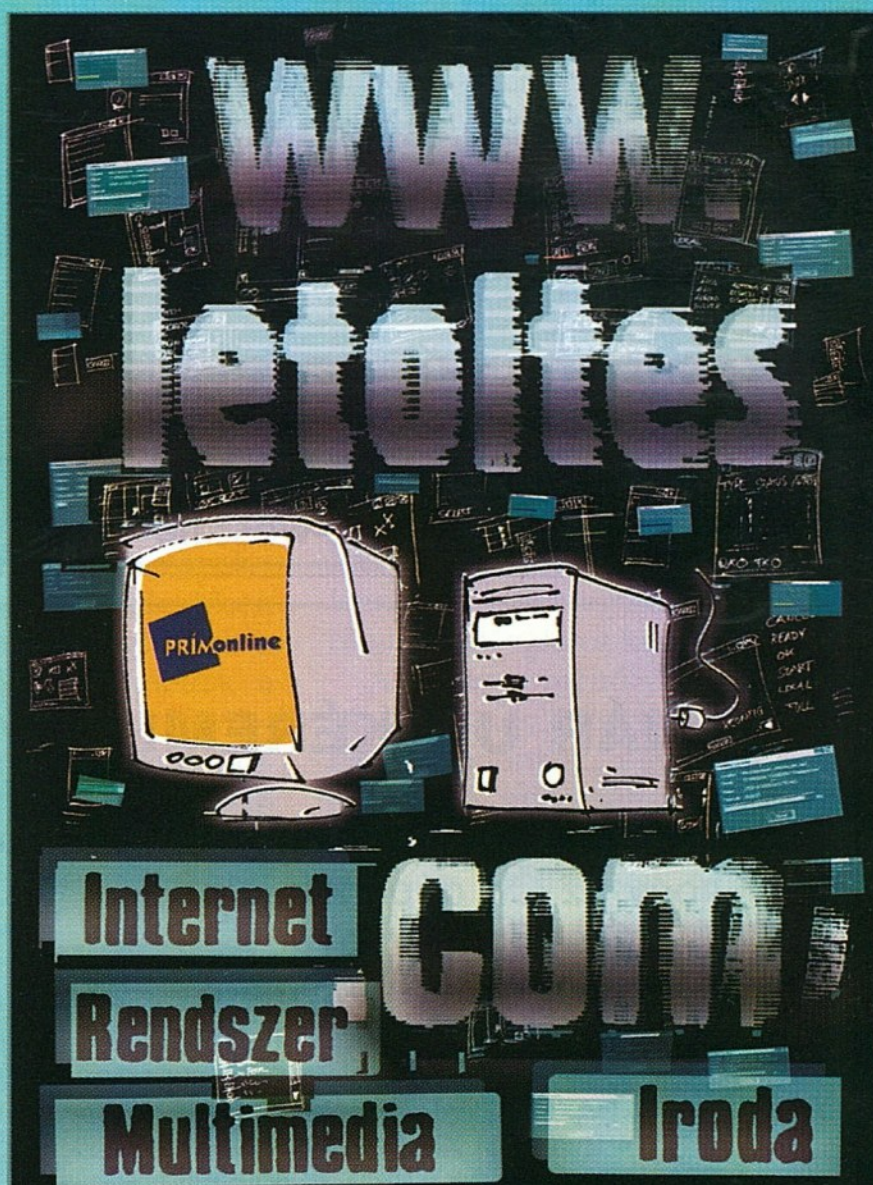
Számítógéppel emberi nyelven

Intelligens szövegkezelés számítógéppel
Szak Kiadó, Bicske, 1999
344 oldal, 2800 Ft









INFORMÁCIÓKÉRÉS: 21 ▼

INFORMÁCIÓKÉRÉS: 23 ▲

INFORMÁCIÓKÉRÉS: 07 ▲

INFORMÁCIÓKÉRÉS: 04 ▲

INFORMÁCIÓKÉRÉS: 12 ▲

INFORMÁCIÓKÉRÉS: 09 ▲

rendkívül gazdag, és tematikájában igen széles területről ad jól átfogó képet. A könyv következő kiadásában bizonyára sok újabb dolog is szerepelni fog, és sort kerítenek a szerzők a hibák lenyesegetésére is.

Megszívlelendő megjegyzések

1. A szerzők számos témát csak futólag említenek meg, de a könyv szövegében nem tartották szükségesnek részletezni ezeket, vagy a többlet a kézirat lezárása miatt már nem kerülhetett bele. Megemlékeznek például egy újonnan készült számítógépes programról, amely ékezetek nélkül írt magyar szövegek ékezetesítését végzi el, lehetőleg kivédve a többértelműség által okozott problémákat.

2. Elég részletesen foglalkoznak a szerzők a rendezés témakörével. Megemlítik például, hogy kétféle rendezési szabály létezik: a könyvtári és az akadémiai — de éppen csak megemlítik az előbbit. Ez önmagában is olyan kettősséget jelent, amelyen szerintem nem szabad könnyedén átsiklani. Márcsak azért sem, mert lexikonaink, enciklopédiáink általában egyjegyű betűk szerint vannak rendezve, a rendezés módja pedig, mint tudjuk, a keresést is alapvetően meghatározza. Hol a helye például a kötőjelnek? A szóköznek? Mi legyen a több szóból álló címszavak rendezésével? Hol keressük például a Britannicában az „El Salvador" címszót? Az „Elba" előtt vagy után? Mi legyen a régies írású nevekkel? A Cholnoky név például a c vagy a cs betűhöz igazodjon? És mit tegyen az a számítástechnikus, aki különböző rendszerek fejlesztése során efféle problémákkal találkozik?

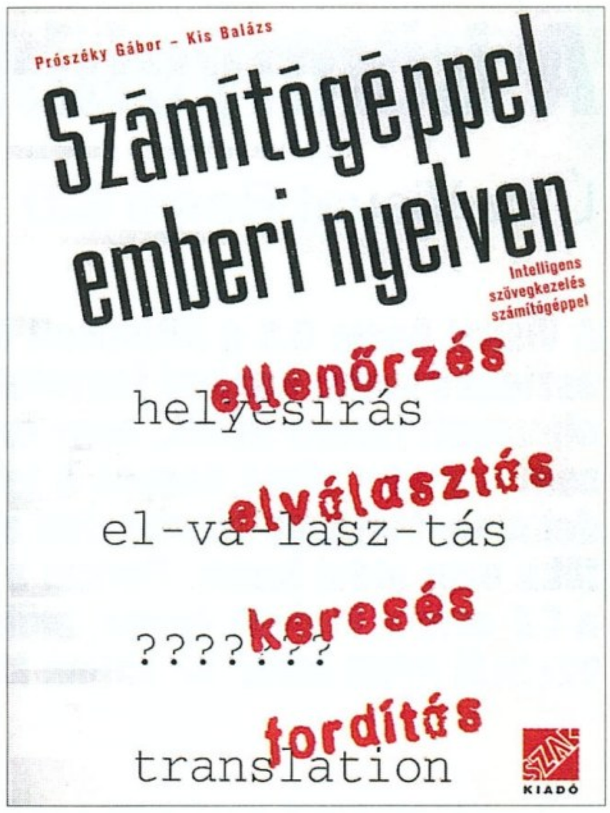
Szótárak esetében is újragondolást kívánhatnak a részleges formalizáláson alapuló akadémiai előírások a többjegyű betűk rendezésével kapcsolatban. A magyar – idegen nyelvű szótárak használói között külföldiek is vannak. Miért kötelesek ők előre tudni egy szó megkereséséhez, hogy mi a helyes kiejtése? Hogy a „község", a „malacsült" vagy a „mézsör" szavakat hogyan kell tagolni? Hiszen még a magyarok sem mindig tudják, hogy a „karosszéria" szónak semmi köze a „karos" szóhoz! Egyre inkább számítógépesedő világunkban érdemes-e még fenntartani azt a rendezési elvet, hogy a „meszel" megelőzi a „messzelátó"-t, de a „mesztic" és a „mésztej" csak a hosszú mássalhangzót jelölő „ssz"-es szavak után következhet? Nem jelent-e több zavart, mint előnyt a jelenleg uralkodó kétlelkűség a rendezésben?

3. Megérdemelt volna valamivel bővebb részletezést a transzliteráció megfordíthatóságának a kérdése is. Mint követelményt említik a szerzők a kölcsönösen egyértelmű kódolást, de nem térnek ki rá, hogy milyen betűket mos össze (fonetikus alapon) az orosz szavak magyar átírása, és hogy a szabványosnak mondott angol átírás visszakódolása milyen problémákat vet fel.

4. Némileg félrevezető az az információ, amit a tezauruszokkal kapcsolatban adnak a szerzők. Ez azért is zavaró, mert maguk állapítják meg a szinonimaszótár és a tezaurusz fogalmának bevezetésekor, hogy „az összehasonlítás [. . .] mindkét szótárfajta pontos definiálásával a legkönnyebb". A tezaurusz pontos definíciója azonban kimaradt a könyvből — maguk a szerzők is legalább kétféle értelemben használják. Az első helyen (pp. 159–162) a ma már „klasszikus" Roget-féle értelemben, mint fogalmi körök szerint rendezett szinonimaszótárt, a kereséssel kapcsolatban azonban (p. 213 etc.) már bizonyára a relációs kapcsolatokkal kiegészített ún. információs tezauruszokra gondolnak.

Roget volt az, aki 18 éves korában kigondolta, majd 73 éves korára megvalósította az angol nyelv szavainak fogalmi struktúrába rendezését, ezer ágra bomló laza szinonimalisták formájában. Még életében 28 kiadást ért meg a nagy mű, s azóta is számtalan kiadás készült belőle, alapos revízió és kibővítés után. Roget tezauruszkoncepciójának az volt az elsődleges célja, hogy segítsen színesebbé tenni íróknak, lektoroknak, bárkinek az írott szövegek készítését. Pontosan az ellenkező céllal készültek a XX. század 60-as éveitől az információs tezauruszok. Elsősorban az indexelést és az információkeresést akarták szolgálni, hogy kiszűrjék a természetes nyelvek színességét, lecsupaszítsák olyan szürke információs nyelvre, amelyben keresgélve jelentősen megnő a találati valószínűség. Ezek a tezauruszok kötelezően feltüntették a szócikkek között fennálló alá- és fölérendeltségi viszonyokat, a szinonimarelációt (a preferált csoportképviselő megjelölésével), és ha mást nem, legalább a címszóval lazább értelmi kapcsolatban álló, ún. rokon fogalmak címszavait. A természetes nyelv Roget-féle laza struktúrájával szemben tehát itt egy-egy szakterület leginkább releváns, jellemző szavaiból építettek fel kimondottan informatikai célokat szolgáló keresőnyelvet.

Való igaz, hogy a teljes szövegű keresőrendszerek megjelenésével, és a



számítógépek hatékonyságának növekedésével némileg elmosódott a különbség a kétféle tezaurusz között. Ma már számítástechnikai segédeszközként is lehet használni a Roget-féle koncepció alapján készült szótárakat, a szöveg színesítésére és a keresés hatékonyabbá tételére is. A fogalmi relációk elnagyolt bemutatása azonban hiányérzetet kelt az olvasóban.

5. A könyvben nehezen azonosíthatók a szövegen belüli hivatkozások, mivel ehhez a fejezetek előtti tartalomjegyzékben megadottnál részletesebb bontásra lenne szükség. A felhasznált irodalomra szöveg közben szerző+évszám hivatkozást használnak a szerzők. Ez azonban nem egyértelmű, ahol az illetőnek egy évben több művét is felsorolja az irodalomjegyzék. (A gyakran említett [Prószéky, 1998] pedig egyáltalán nem szerepel az irodalomjegyzékben.) Egyébként is jó lenne a szöveget itt-ott dúsítani máshonnan vett irodalommal is.

6. Meggondolandó, hogy a szövegben említett MoBiDic termékekről nem lenne-e érdemes egy függelékben tematikai összefoglalót készíteni.

A könyv méltó bemutatkozása a MorphoLogic csoportnak, úgy is mint kutatócsoportnak, úgy is mint értékes szoftverek gyártójának. A legszebb elismerés az lenne a szerzőknek, ha példájuk és útmutatásaik nyomán mások is bedolgoznák magukat az intelligens számítógépes szövegkezelésbe. Komoly érdemük: sikerült bemutatniuk a terület szépségein kívül a megoldásra váró problémákat is, melyek gazdagsága remélhetően sokakat fog újabb problémák megfogalmazására és újabb ötletek megvalósítására ösztönözni.

Vargha Dénes

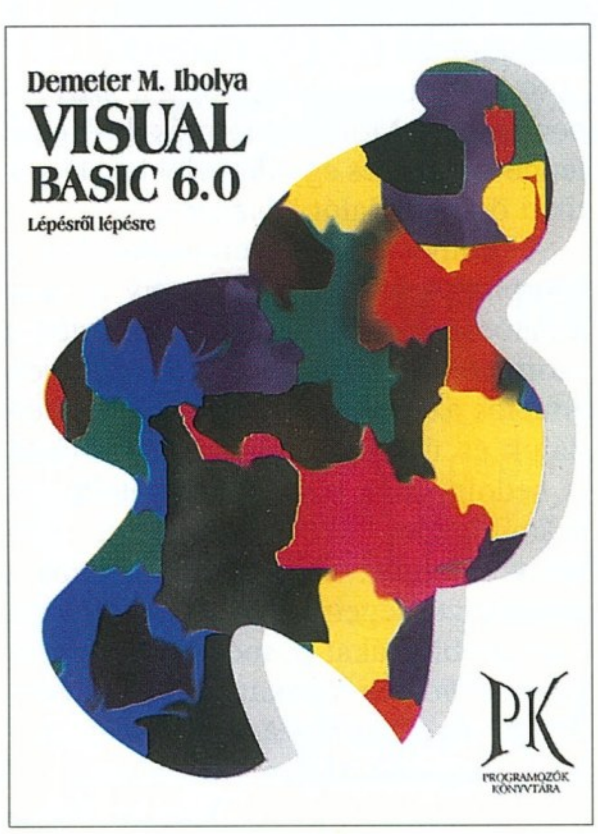
A könnyebbik út

Újra Visual Basic 6.0

A Visual Basic 6.0 a Microsoft Visual Studio rendszerének oszlopos tagja. Kezdeti fogyatékosságait már kinőtte, és a Microsoft javára írható, hogy ezúttal már az információkat sem sajnálja olyan nagyon a felhasználóktól. Két CD-nyi dokumentációban töméntelen anyag fér el. Kinyomtatva több ezer oldal lenne. Persze a gyakorlati felhasználáshoz a CD nem a legjobb forma, működés közben sokkal több hasznát lehet venni az online help rendszernek.

Becsülendő újítás, hogy a Microsoft kontextusfüggő (tartalomérzékeny) súgót fejlesztett ki a VB 6.0-hoz, amely a Help menü hívása nélkül is elérhető. Tartalomérzékeny súgó tartozik a fejlesztőrendszer legfontosabb részeihez: minden ablakához, az Eszköztár (Toolbar) elemeihez, a formán elhelyezhető kontrollokhoz, a Form vagy Documentum objektum további objektumaihoz, a tulajdonságokhoz a tulajdonságablakban, az eseményprocedúrákhoz a kódablakban, a VB kulcsszavaihoz (kifejezések, függvények, tulajdonságok, metódusok, események, speciális objektumok elnevezéseihez) és a hibaüzenetekhez. A CD-khez már csak akkor érdemes fordulni, ha bővebb magyarázatra van szükségünk.

A Microsoft Press a Park Kiadóval összefogva magyarul is megjelentetett már a VB 6.0-ról egy vastag kézikönyvet (írtunk is róla 1999. augusztusi számunkban). Sok tekintetben azonban a magyar olvasók igényeinek jobban megfelel Demeter M. Ibolya könyve az új változatról. A szerző oktatja is a VB-ben való programozást, így jobban ismeri, milyen problémák szoktak felmerülni a hallgatóság körében. Ráadásul a szerző az előző változatokról is írt több könyvet, így fejlődésében is ismeri a témát.



Alcímének megfelelően a könyv lépésről lépésre vezeti be olvasóit a VB programozás rejtelmeibe. Igen sok példán keresztül mutatja be azokat a kisebb-nagyobb trükköket, amelyek elsajátítása nyomán a kezdő programozó is hamarosan otthonosan mozog a vizuális programozás fejlesztői környezetében. Frappáns példái hellyel-közzel a professzionális programozókból is kiváltják az „aha!" effektust. A könyv bőséges illusztrációi és táblázatos összefoglalói jól áttekinthetővé teszik az anyagot.

A Microsoft régi szokásához híven most is több bőrt igyekszik lehúzni egy rókáról. A „kezdő", „középhaladó" és „haladó" kategóriákat ezúttal is a „Learning", „Professional" és az „Enterprise Edition" elnevezésekkel teszi nehezebben felismerhetővé. A „Learning Edition" valóban annyira a kezdők számára készült, hogy igényesebb programozóknak még tanulásra sem ajánlható. Az adateléréshez szükséges dolgok teljesen hiányoznak belőle (Data Environment, Integrált adatbázistervezési eszközök, Data View ablak, Data Objectum és kibővített Data Form varázsló, kiterjesztett adatkapcsolások, adatforrások létrehozása, adatbázison alapuló jelentések készítése stb.). De hiányzik a Toolbar Control, sok fontos ActiveX kontroll, mint a DataRepeater vagy a ListView, és több internetes lehetőség (DHTML és IIS-alkalmazások, Explorer 4.x támogatás az ActiveX dokumentumokhoz). Lényegesen kisebb a különbség a Professional és az Enterprise változatok között (a legnagyobb különbség az integrált adatbázistervezési eszközök és az SQL Editor hiánya az előbbiben).

Sajnálatos, hogy a könyvbe nem (vagy csak módjával) kerülhettek bele olyan lényeges dolgok, mint az objektumok programozása (ActiveX komponensek, ActiveX tervezők, objektumosztályok), a helpkészítés módszerei, grafikai, egérkezelési és nyomtatási lehetőségek. Érthető módon ezek felvétele erősen felduzzasztotta volna a kötet terjedelmét, ezért a szerző egy második kötet megírását tervezi. Amit viszont elkészített, az igen jó segítség mindazoknak, akik biztonságosan el akarják sajátítani a vizuális programozás műhelytitkait, bele akarnak kóstolni az ízébe. Tagadhatatlan, hogy ez a programozás legkényelmesebb módja, könnyen elsajátítható, és most már a legfontosabb információkat is tálcán kínálják. Arról azonban senki ne feledkezzen meg, hogy 16 bites környezetben már a VB 5.0 verziója sem működött!

A szerző jó stílusban képes átadni az olvasóknak az ismertetett programrendszerekre vonatkozó legfontosabb tudnivalókat, és igen jól válogatta ki szövegéhez a példaanyagot. Nagy kár, hogy a kötet értékét szerkesztési és formai hiányosságok csökkentik. A sorozat tipográfiai konvenciói nem a legügyesebbek, a különböző szintű címek rengetegében inkább csak a tartalomjegyzék segítségével lehet tájékozódni, mert a betűtípusok hierarchiája nem érzékelhető világosan. Az ábrák egy részének minősége nagyon gyenge: olvashatatlanok a szövegeik, felismerhetetlenek az ikonjaik. A tárgymutatóban is sok hiba van, fontos kifejezések kimaradtak, oldalszámok elcsúsztak stb. Mintha elkapkodták volna a könyv kivitelezésének végét, ami egyébként nem jellemző a kimondottan igényes, és kiadványait jól megválogató kiadóra. Most valami homokszem kerülhetett a gépezetbe.

Vargha Dénes

Demeter M. Ibolya:

Visual Basic 6.0

Lépésről lépésre
Panem Kiadó, Programozók Könyvtára, 1999
568 oldal, 2900 Ft

ELŐFIZETÉS

Az 2000/......... számtól kezdődően előfizetem

az Új Alaplap című CD-mellékletes havi számítástechnikai folyóiratot

...... példányban ❑ 1 évre, ❑ 1/2 évre.
Az éves előfizetési díj: 6996,– Ft (Ez az összeg az áfát is tartalmazza.)

❑ **Számlát** kérek (banki átutalással fizetek). ❑ **Befizetési csekket** kérek.

Név: ....

(Cég:) ....

Cím: ....

Irányítószám, helység: ....

Dátum:

/aláírás/

APRÓHIRDETÉS

Kérem, hogy az Új Alaplap következő számának Mikrobazár rovatában az alábbi szövegű apróhirdetést jelentessék meg. (A túloldalon ismertetett feltételeket tudomásul veszem.)

....

....

....

....

....

....

....

....

(Maximális terjedelem 300 betű.)

INFORMÁCIÓKÉRÉS

Az Új Alaplap mostani számában megjelent hirdetések közül az általam itt megjelölt kódszámúakhoz részletesebb információt kérek a hirdető cégektől.

01	02	03	04	05	06	07	08	09	10	11	12
13	14	15	16	17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81	82	83	84
85	86	87	88	89	90	91	92	93	94	95	96

Új Alaplap, 2000. áprilisi szám. Beküldési határidő: 2000. április 30.

Belföldön díjmentesen is feladható

ÚJ ALAPLAP

VI., Dózsa György út 84/b
Postafiók 571
1539 Budapest

Feladáskor kérjük bérmentesíteni!

FELADÓ:

Név:...

Cím:...

Helység: ....

Irányítószám:..

Telefon: ....

❑ A hirdetés egyéni és egyedi jellegű, ezért kérem ingyenes megjelentetését. Kijelentem, hogy annak tartalma nem sérti senki szerzői jogát.

❑ A hirdetés kereskedelmi célt szolgál. Mellékelem a soronként (60 karakterenként) 300 forintnak megfelelő összeg átutalásáról az igazoló szelvény másolatát. A címzett: Új Alaplap, 1539 Budapest, Pf. 571, illetve átutalásnál az OTP 11706016-20788599 számlaszámra.

- - - - - - - - - - - - - - - - -
/aláírás/

Bélyeg helye

ÚJ ALAPLAP

VI., Dózsa György út 84/b
Postafiók 571
1539 Budapest

FELADÓ:

Feladáskor kérjük bérmentesíteni!

a) EGYÉNI

Név: ....

Cím: ....

Helység, ir.sz.: ....

b) CÉGES

Név: ....

Cég: ....

Cím: ....

Helység, ir.sz.: ....

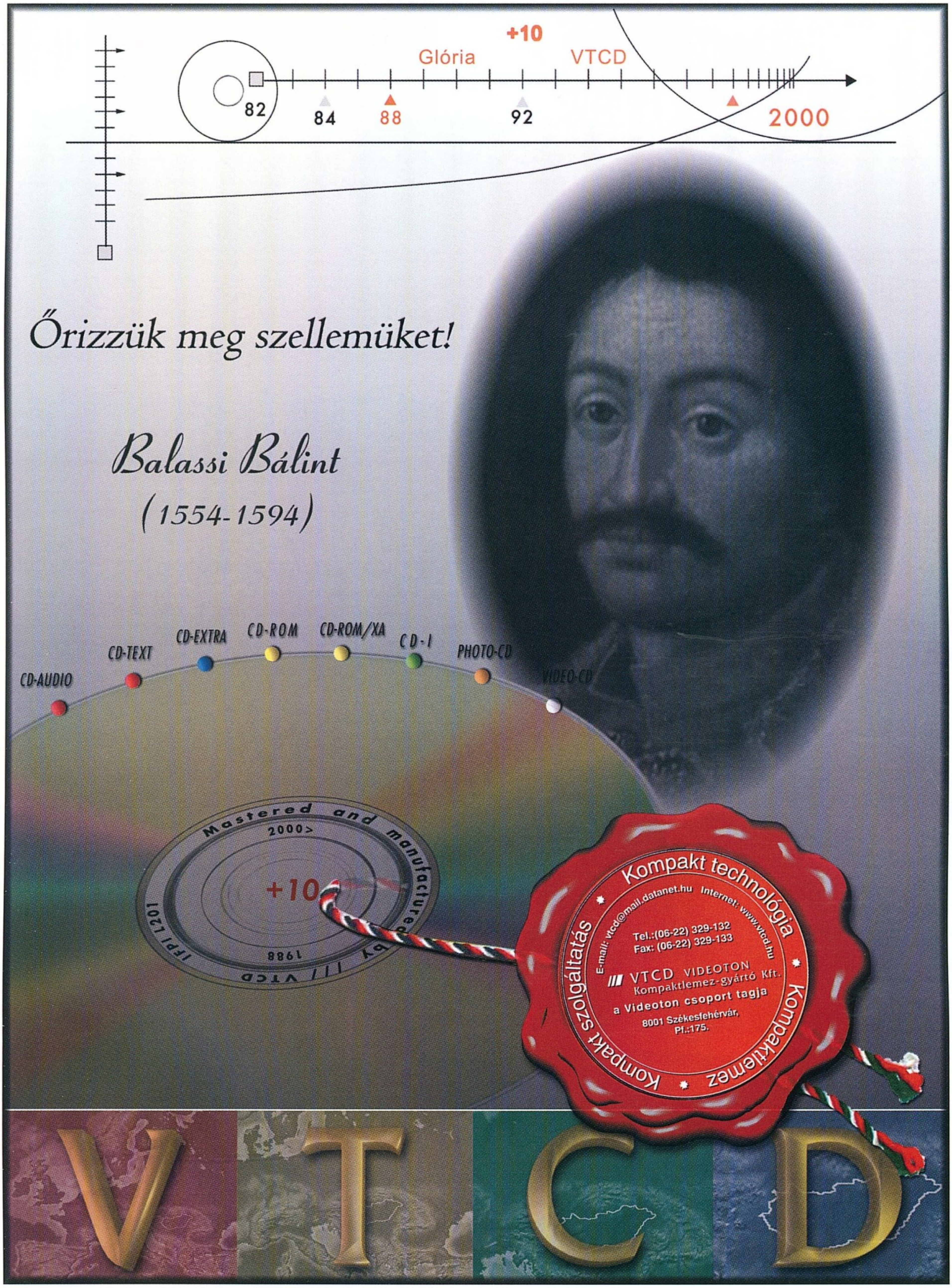
Telefon: ....

- - - - - - - - - - - - - - - - -
/aláírás/

Bélyeg helye

ÚJ ALAPLAP

VI., Dózsa György út 84/b
Postafiók 571
1539 Budapest


+10
Glória
VTCD
82
84
88
92
2000
Őrizzük meg szellemüket!
Balassi Bálint
(1554-1594)
CD-AUDIO
CD-TEXT
CD-EXTRA
CD-ROM
CD-ROM/XA
CD-I
PHOTO-CD
VIDEO-CD
Mastered and manufactured by /// VTCD
2000>
+10
1988
IFPI L201
Kompakt technológia
Kompaktlemez
Kompakt szolgáltatás
E-mail: vtcd@mail.datanet.hu
Internet: www.vtcd.hu
Tel.:(06-22) 329-132
Fax: (06-22) 329-133
VTCD VIDEOTON
Kompaktlemez-gyártó Kft.
a Videoton csoport tagja
8001 Székesfehérvár,
Pf.:175.
VTCD

HP Vevőszolgálat: 382-1111 www.hp.hu





Megtévesztően valósághű.

Ugye szinte tart attól, hogy a szemébe pattanhatnak ezek a szikrák, annyira valósághűek? Megnyugodhat, ez csak egy másolat, amely a létező legfejlettebb tintasugaras nyomtatási eljárással készült. Ezt nyújtja Önnek a Hewlett-Packard forradalmi, PhotoREt precíziós technológiája.

Képes akár 29 tintacseppet elhelyezni minden egyes mikroszkópikus ponton, ráadásul mindezt rendkívül gyorsan. Bármennyire sietős is az Ön dolga, mindig egyedülálló fotóminőségű képeket fog kapni. Ezt senki nem csinálja jobban. Legföljebb maga a valóság.



HP PhotoSmart és DeskJet nyomtatók

invent